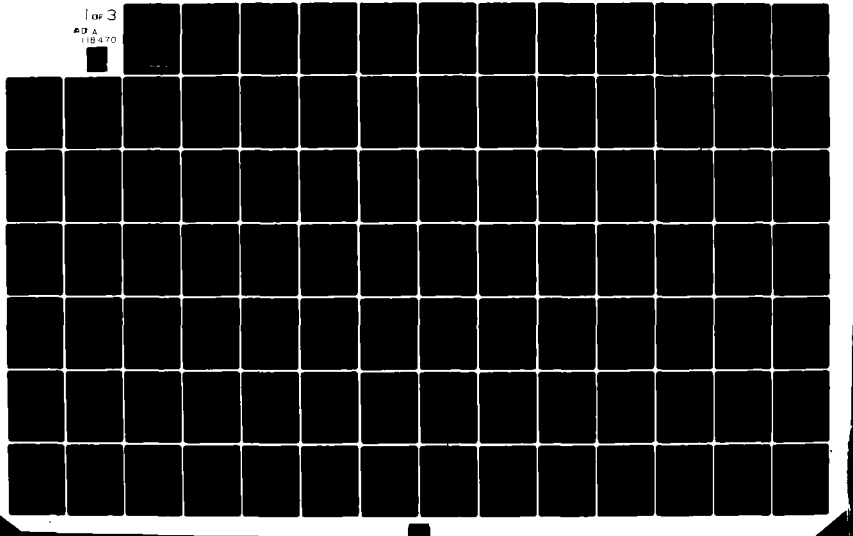
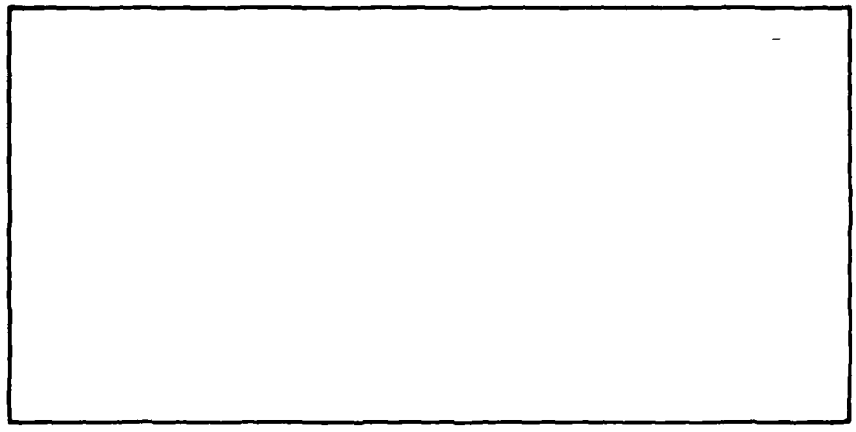
AD-A118 470 TRACOR APPLIED SCIENCES AUSTIN TX F/G 15/4
HYBRID TRACKING ALGORITHM IMPROVEMENTS AND CLUSTER ANALYSIS MET--ETC(U)
FEB 82 D COOPER, G CORSER, T WILSON N00014-78-C-0670
UNCLASSIFIED TRACOR-T82-AU-9054-U NL





Tracor Applied Sciences

Tracor, Inc 6500 Tracor Lane Austin Texas 78721

Contract N00014-78-C-0670
Tracor Project 036-013
Document No. T82-AU-9054-U

(12)

FINAL REPORT: HYBRID TRACKING ALGORITHM IMPROVEMENTS AND CLUSTER ANALYSIS METHODS

By

Don Cooper, Glenn Corser, and Thomas Wilson

Submitted to

Scientific Officer
Statistics and Probability Program
Office of Naval Research
800 North Quincy Street
Arlington, Virginia 22207

Attention: James G. Smith
Code 411 (SP)

Submitted by:

Don Cooper

Don Cooper
Project Engineer

Approved by:

Frank C. Harris

Frank C. Harris
Department Director
System Modeling Department

26 February 1982

Tracor Applied Sciences

Tracor, Inc. 6500 Tracor Lane Austin, Texas 78721 Telephone 512. 926 2800

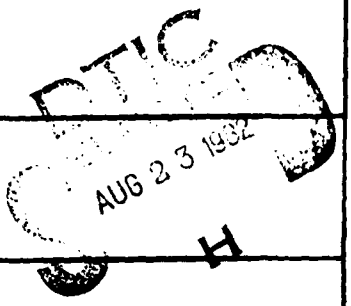
UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER T82-AU-9054-U	2. GOVT ACCESSION NO. AD-A118 470	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) FINAL REPORT, HYBRID TRACKING ALGORITHM IMPROVEMENTS AND CLUSTER ANALYSIS METHODS		5. TYPE OF REPORT & PERIOD COVERED Final Report 1 Jan 1981 - 26 Feb 1982
		6. PERFORMING ORG. REPORT NUMBER T82-AU-9054-U
7. AUTHOR(s) Don Cooper, Glenn Corser and Thomas Wilson		8. CONTRACT OR GRANT NUMBER(s) N0014-78-C-0670
9. PERFORMING ORGANIZATION NAME AND ADDRESS Tracor Applied Sciences 6500 Tracor Lane Austin, TX 78721		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 65152N, R0145-TW, NR 274-299
11. CONTROLLING OFFICE NAME AND ADDRESS Probability and Statistics Program Office of Naval Research 800 North Quincy St.; Arlington, VA 22207		12. REPORT DATE 26 February 1982 13. NUMBER OF PAGES Page i - A-52
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release: distribution unlimited



17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Target Tracking	Multi-Target Tracking	Response Surface Methodology
Localization	Data Sorting	
Hybrid Tracker	Cluster Analysis	Experimental Design

20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report details three areas of research undertaken during the past contractual year. First, detailed explanations are provided concerning the modifications made to Tracor's Hybrid Tracking Algorithm(HTA)to improve its overall tracking performance. Second, a summary is pro ded of Tracor's preliminary investigation into the use of cluster analysis techniques to sort data in the multi-target tracking problem. Finally, a description is provided of an experimental design technique known as response surface methodology. This design is used to quantify HTA's tracking performance as functions of certain data gathering parameters.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

UNCLASSIFIED

EXECUTIVE SUMMARY

This report describes the current findings and status of Tracor's ongoing investigation of the multiple target tracking problem. In particular, we have concentrated on the problem of tracking multiple targets with data gathered by distributed, passive acoustic sonobuoys. In this study, the multiple target tracking problem has been initially divided into two separate tasks: (1) the development of an efficient, highly accurate, single target tracking algorithm; and (2) the development of a non-parametric data sorting technique for separating a sonobuoy's multiple target data stream into sets of individual target data. Also included as an appendix is a detailed discussion of an experimental design technique known as Response Surface Methodology (RSM) that was used to quantify the single target tracking algorithm's response to variations in signal gathering and signal processing parameters.

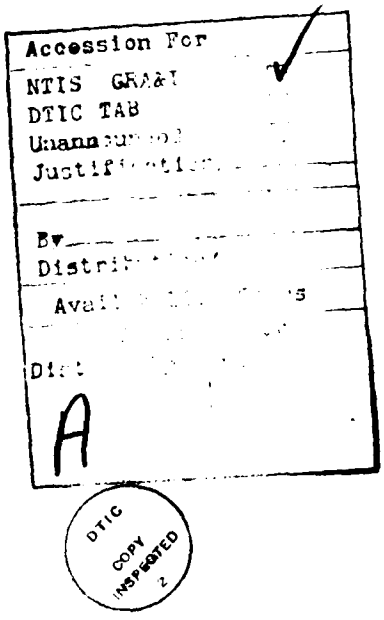
In the past, Tracor has developed both a Hybrid Tracking Algorithm (HTA) and a Sequential algorithm to perform the single target tracking task. The former algorithm is known as a hybrid algorithm because it uses a batch tracker to initialize the tracking solution, and after the track has been successfully initialized, it automatically switches to a sequential tracker to continue updating the target's trajectory. The current investigation has sought to improve further both trackers' performances, and with that objective the following modifications were made to their prior designs.

(1) The Sequential's initializer has been modified to use a Standard Kalman Filter plus a one-dimensional, numerical search

technique to reduce the number of iterations needed for the totally sequential tracker to initialize a track.

(2) A batch-type, initial guess algorithm has been developed which uses the initial frequency and bearing measurements from two or more sensors in a "crossed-bearings, crossed frequencies" technique to provide reasonable guesses of the target's position and velocity to both the HTA's and the Sequential's initializer.

(3) The target's dynamic acceleration model has been changed to a normal-tangential (or along track-across track) acceleration model to better describe possible target maneuvers.



(4) A mobile sensor model has been added to the tracker to allow possible sensor motions created by either sensor drift or by mobile sensor platforms.

(5) Finally, range, time-difference of arrival, Doppler ratio, and Doppler difference measurements have been added to the tracker to augment the frequency and bearing measurement models that could be used initially.

In order to extend the HTA or other single target trackers into the area of multiple target tracking, Tracor initiated an investigation into the possibility of using

cluster analysis techniques to sort multiple target data at the individual sonobuoy level. Cluster analytic methods form a branch of numerical taxonomy which can be used to search quantitatively for natural groups or clusters within a set of objects which have been described by an arbitrary set of descriptive attributes. This initial investigation has shown the application of cluster analysis methods to be a potentially feasible means for solving the multiple target data sorting problem. From our cluster analysis investigation, the following configuration for processing the data has yielded the best results:

(1) Four attributes have been used to describe each of the multiple target measurements:
   (a) Measurement time tag
   (b) Frequency estimate
   (c) Cosine of the bearing estimate
   (d) Sine of the bearing estimate

(2) Each of these four attributes were normalized to lie between 0 and 1.

(3) Euclidean distances were used to generate a resemblance matrix of dissimilarity coefficients between each measurement pair.

(4) Hierarchical, single-linkage clustering algorithms have been shown to be the most effective for sorting the data.

The cluster analysis techniques have been shown to be effective at performing the following tasks:

(1) Outlier identification for single target data sets.

(2) Sorting sets of multiple target data into individual target data sets.

(3) Sorting multiple signals from ambient noise in simulated DIFAR power spectra.

Unfortunately, at their present stage of development, the cluster analysis techniques studied require that some a priori knowledge of the data be available before their results can be properly interpreted. The clustering results are currently output as tree diagrams and require the analyst to carefully study the results to pick the optimal set of clusters. However, it is felt that with further investigation and development, these clustering techniques can be automated so that "intelligent" operator interpretations of the results will not be required. Then, these techniques can be used in real systems.

For both the single target tracking and the data sorting investigations, a non-Guassian, DIFAR data generation model was used to simulate the narrowband frequency and bearing measurements. This DIFAR simulation models variations in the signal-to-noise ratio (SNR) of the signal received by the sonobuoy that are caused by propagation losses, smearing losses, and random variations in both the target's radiated levels and in ambient noise levels. The error distributions from this simulation program are non-Gaussian, so more realistic investigations of tracker performance and data sorting performance can be undertaken than would be possible with a simpler Gaussian model.

Finally, this report contains a detailed discussion of the RSM techniques used to quantitatively analyze HTA's responses to variations in data gathering and data processing parameters. Specifically, this study investigated how changes in target signal strength, changes in sonobuoy baseline distances, and changes in processor integration time for generating frequency and bearing estimates affect the overall tracking performance of Tracor's HTA. This investigation was initiated not only to characterize HTA's tracking performance, but it was also meant to show how RSM or other experimental design techniques can be used to quantify various algorithms' responses to variations in key parameters so that they can be more effectively evaluated or compared against other possible alternatives.

TABLE OF CONTENTS

TABLE OF CONTENTS -- Continued

TABLE OF CONTENTS -- Continued

TABLE OF CONTENTS -- Concluded

LIST OF IULLUSTRATIONS

LIST OF IULLUSTRATIONS -- Continued

LIST OF IULLUSTRATIONS -- Concluded

LIST OF TABLES

1.0 INTRODUCTION

For many anti-submarine warfare (ASW) encounters, the U.S. Navy is very concerned with the problem of detecting, classifying, and tracking underwater submarines from data gathered by passive sonobuoy patterns. One area where Tracor has been heavily involved is the area of target tracking or localization.

Tracor has developed several target tracking algorithms, but they have dealt only with the question of tracking one target from single target data gathered by various acoustic signal processors. Another problem of great importance to the Navy concerns the questions of detecting, classifying, and tracking multiple targets when they are observed in tracking environments.

At Tracor, this multiple target tracking problem has initially been divided into two distinct tasks. First, a highly accurate, single target tracking algorithm has been developed to determine current capabilities for localizing single targets. Separate from this development, an investigation has begun into the question of sorting and classifying passive, multiple target data into sets of data for each target. As this investigation continues, these two questions will have to be considered together along with many other problems. However, for now, only the development of an accurate and reliable single target tracking algorithm along with an initial approach for sorting data in the multiple target problem have been considered. The following subsections summarize the work performed under the current contract in these two basic task areas.

1.1 Hybrid and Sequential Algorithms' Modifications

Two different algorithms have been developed in the past to track one target. One algorithm, the Hybrid Tracking Algorithm, uses a weighted, least-squares, "batch " procedure to initialize the track and switches to an Extended Kalman Filter to continue tracking after the tracker has been successfully initialized. The other, a Sequential algorithm, has also been developed. This algorithm used an Extended Kalman Filter to both initialize the tracker and to continue tracking the target after successful initialization has been achieved. Both algorithms performed well, but it was felt that both could be modified to improve their tracking accuracies, range of applications, and tracker initialization characteristics. Following is the list of modifications implemented to improve these tracking algorithms.

(1) The sequential initializer was improved by using a Standard Kalman Filter plus a numerical, one-dimensional search technique to find the optimal initial state for the target. This procedure proved to be successful because fewer data points and fewer iterations (relative to the old design) were required for the new sequential initializer to converge onto an acceptable set of initial conditions for the target.

(2) An inital guess algorithm was developed to use the sonobuoys' data to provide a better initial guess of the target's state. This algorithm is used by both the Sequential's and Hybrid's initializers.

Previously, an arbitrary point was picked and the initializers sought to change this guess until a suitable set of initial conditions was found. Now the initial guess algorithm uses overlapping frequency and bearing measurements from at least two different sonobuoys in a "crossed-bearing, crossed-frequency" procedure to generate reasonable least squares estimates of the target's position and velocity. This initial guess procedure has proved to be more accurate and faster than the previous method.

(3) The target's motion model has been changed from a Cartesian acceleration model to a normal-tangential model. In this context, the tangential direction is defined to lie parallel to the target's course heading and the normal direction lies perpendicular to this course heading. This new acceleration model has been shown to be better than the Cartesian one for modeling both maneuvering and non-maneuvering target trajectories.

(4) A mobile sensor model has replaced the old stationary model used for positioning the sonobuoys. This allows the tracking algorithms to process data from drifting sensors as well as data from mobile sensors such as hull-mounted and towed-array systems.

(5) New data models were also added to the trackers' measurement models. In the past, only frequency and bearing measurements have been used for localization by the Hybrid and Sequential trackers. Now range, time-difference of arrival, Doppler ratio, and Doppler difference measurements can also be processed by these trackers.

1.2 Simulation of Single Target Data

The data generation program used for these studies simulated non-Gaussian, frequency and bearing estimates for narrowband signals. This program simulated a comb filter bank followed by a square law detector to generate an omni-directional power versus frequency spectrum. A frequency estimate was obtained from this power spectrum by a peak-picking procedure which selects the single comb filter bin in the spectrum that contained the most omnidirectional power. After the frequency estimate was obtained, an arctangent estimator used the simulated x and y channel power associated with this chosen frequency bin to generate a bearing estimate. If the signal strength of these estimates exceeded a set threshold level, the estimates were accepted; if not, no measurements were passed to the tracking algorithm.

1.3 Simulation of Multiple Target Data

To study the multiple target sorting problem, a suitable data base had to be developed. A single target data simulation program already existed, so it was decided to build programs that could merge several different single target data sets into one simulated multiple target data set. Initially,

one program was developed that merged the output frequency and bearing estimates for each simulated target into one set of measurements for all of the targets. However, some theoretical difficulties were encountered with this approach, so a second program was developed. The second program merged the power versus frequency spectrum for each of the targets into one single spectrum. The data sorting techniques were then employed at this multiple target, power spectrum level to perform the sorting task. Data from both simulations were used in the multiple target, data sorting investigation.

1.4 Multiple Target Data Sorting

Cluster analysis techniques have been chosen for this initial investigation of the data sorting problem associated with tracking multiple targets. This technique is used in numerical taxonomy to search for natural groups or clusters from a set of objects which have been described by an arbitrary set of descriptive attributes. No *a priori* functional form or conditional relationship is assumed for the objects and their attributes. Instead, the observer must only pick the set of attributes that are to be used to describe the objects and the clustering algorithms then search for natural groupings of the objects based on these attributes. Extensive development and use of these clustering techniques can be found in the anthropological, biological, and social sciences.

Five different non-overlapping, hierarchical clustering algorithms have been investigated. In addition, five different data normalization techniques were studied, as well as seven different methods for generating similarity-dissimilarity coefficients. All of these techniques are described in detail in Section 5. Based on the results

obtained, the following conclusions have been formed from this investigation:

(1) Attribute data should be normalized so that the range of values lies between 0 and 1.

(2) The average Euclidean distance dissimilarity coefficient proved to be the most useful for generating the resemblance matrix for the data sorting problem.

(3) The single linkage clustering methods yielded the best results for the data sorting problem.

To sort the data, a set of attributes must be used to describe the objects of interest. For the passive data simulated in this investigation, the following set of attributes was found to be the most useful for sorting the multiple target measurements.

(1) Time tag of the measurement estimates

(2) Frequency estimates

(3) Cosine of the bearing estimate

(4) Sine of the bearing estimate

When the multiple target data have been described with these attributes, the single linkage clustering algorithm has been successful in performing the following functions:

(1) Identifying outliers to be removed from the data set.

(2) Sorting multiple target measurements into sets of individual target data.

(3) Sorting multiple signals from ambient noise in simulated power spectra data.

These results have been quite encouraging. At present, however, the algorithm is cumbersome and requires considerable operator interaction. The difficulties appear to be traceable to attribute normalization problems. The normalization used for this study permitted successful clustering of acoustic data, but a priori knowledge of the data was required to set appropriate dissimilarity coefficient thresholds to properly define target clusters. Nonetheless, valid data clustering was demonstrated and it appears very promising that efficient, automatic algorithms based on cluster analysis can be developed to sort acoustic data from multiple targets.

1.5 Hybrid Algorithm Sensitivity Study

A study which used Response Surface Methodology (RSM) techniques was initiated to quantify the tracking response of the Hybrid algorithm to external factors such as sonobuoy separation distance, target signal strength, and data integration time. The RSM algorithms fit a polynomial hypersurface in a classical least squares sense to the data obtained from a chosen test design. After a suitable least squares solution has been found, one can then analytically solve for the extremum of this fitted surface. The analyst may also perform eigenvalue analysis to determine whether the extremum

is a minimum, a maximum, or a saddlepoint and may also search for the principle axes to determine directions of maximum and minimum change. Finally, with RSM techniques, one can plot the response surface and its associated contour plot to visually investigate operating range trade-offs. From this analysis, one can determine the optimal operating conditions for a process, as well as compare the response of one process against another process (for instance, the Hybrid's tracking response versus the Maximum Likelihood Estimator's tracking response). Results of this RSM analyis of the Hybrid tracker's response to variations in certain data processing parameters are presented in Appendix A.

1.6 Report Organization

The remainder of this report presents detailed information on the work summarized above. Section 2 describes the modifications made to the Hybrid and Sequential algorithms to improve their tracking performances. Section 3 contains a detailed description of the simulated DIFAR model used to generate single target data. The two techniques for simulating multi-target data are found in Section 4. Detailed descriptions of cluster analysis techniques and of their applications to the multi-target problem are provided in Section 5. Section 6 suggests some recommended research tasks for future investigation, and Section 7 presents a list of references. Finally, the results of the RSM sensitivity analysis of Hybrid's tracking performance are furnished in Appendix A.

2.0 IMPROVEMENTS TO THE TWO TRACKING ALGORITHMS

Under this contract, several modifications were made to both the Sequential and Hybrid trackers to improve their overall effectiveness. Efforts were made to improve the track initialization characteristics for both trackers with special interest taken in improving the Sequential's initializer. Secondly, a new acceleration model was developed to improve both algorithms tracking performance for maneuvering and non-maneuvering trajectories. Lastly, new measurement types and a new mobile sensor model were added to the trackers to increase the possible application areas for both Trackers. Descriptions of these modifications and their ensuing affect on tracker performance are found in this section.

2.1 Improved Sequential Initialization Algorithm

One task in this study sought to improve the Sequential's initializer in an effort to make it more competitive with the Hybrid. Previous results (Reference 1) have shown that the Hybrid outperformed the Sequential in initializing the target tracks from the initial measurement data. After initialization, both trackers' performance was essentially equal. The Hybrid initializer utilized a "batch" filter along with a numerical, one-dimensional search procedure to produce initial state estimates for the target. The Sequential algorithm used an Extended Kalman Filter (EKF) and an iteration scheme to produce estimates of the target's initial state. It was felt that the Sequential's initializer could be greatly improved by replacing the EKF with a Standard Kalman Filter (SKF) and augmenting it with the same one-dimensional search procedure used by the Hybrid's batch initializer. This new Sequential initialization technique has

been developed and incorporated in the tracker. It has proved to be very successful and has made this algorithm more competitive with the Hybrid.

The new Sequential initialization algorithm is a U-D Covariance Factorization of the SKF (Reference 2). Like the batch initializer, the Sequential initializer uses the tracking filter to find a search direction that minimizes the sum of squares of the measurement residuals. The algorithm used to estimate the search direction is given below.

(a) Provide an initial guess for the target's initial state vector x_0, state covariance matrix P_0 and state noise covariance Q.

(b) Decompose P_0 into factors D_0^{-1} and U_0^{-1}.

(c) Initialize the measurement to k = 0. Set the initial search direction to $s_0 = 0$.

(d) Set the measurement counter to k = k + 1 and get a measurement set t_k, y_k.

(e) Solve the following differential equations

$$\dot{x} = f(x,t)$$

where f(x,t) is the target's motion model written as a system of first order equations.

$$\dot{s} = As$$

where

$$A = \frac{\partial f(x,t)}{\partial x}$$

$$\frac{d}{dt}\left(U^{-1}D^{-1}U^{-T}\right) = \dot{U}^{-1}D^{-1}U^{-T}+U^{-1}\dot{D}^{-1}U^{-T}+U^{-1}D^{-1}\dot{U}^{-T}$$

$$= AU^{-1}D^{-1}U^{-T}+U^{-1}D^{-1}U^{-T}A^{T}+Q$$

with initial conditions

$$x = \hat{x}_{k-1}$$

$$s = \hat{s}_{k-1}$$

$$D^{-1} = \hat{D}^{-1}_{k-1}$$

$$U^{-1} = \hat{U}^{-1}_{k-1}$$

(f) Compute the following relationships

$$r_k = y_k - g(x_k, t_k)$$

where $g(x_k, t_k)$ is the computed measurement model at time t_k

$$h_k = \left.\frac{\partial g(x,t)}{\partial x}\right| t = t_k$$

$$v_k = U_k^{-T} h_k$$

$$\beta_k = R_k + v_k D_k^{-1} v_k$$

$$\hat{s}_k = s_k + U_k^{-1} D_k^{-1} v_k \left(\frac{r_k - h_k s_k}{\beta_k}\right), \text{ and}$$

$$\hat{U}_k^{-1}\hat{D}_k^{-1}\hat{U}_k^{-T} = U_k^{-1}\left(D_k^{-1} - D_k^{-1}\frac{v_k v_k^{T}}{\beta_k} D_k^{-1}\right) U_k^{-T}$$

(g) Find α_k, such that

$$\hat{x}_k = x_k + \alpha_k \hat{s}_k$$

is the estimate of the state that minimizes the sum of squares of all measurement residuals. Use a quadratic search procedure to find α.

(h) If the sum of squares from this iteration is within a specified tolerance of the sum of squares from the previous iteration, then the algorithm has converged. Therefore, go to (j).

(i) The algorithm did not converge. Therefore, set x_0 to the initial state vector that satisfied (g) and increase the measurement set. Then go to (d).

(j) Conduct the Modified Gallant Test, described in Reference 1 to determine the need to switch to the EKF target tracking algorithm. If the test is passed, then switch. If not, then go to (d).

Initial parameters are required for the state vector x_0, state covariance matrix P_0, and the state noise covariance Q. Provided that "reasonable" values of these parameters are specified, then this sequential algorithm will converge in the same number of iterations as the batch and to

approximately the same values. The next section outlines an algorithm for obtaining values for x_0 and P_0. Values for Q are still user determined.

2.2 Simultaneous Bearing and Frequency Initial Conditions Algorithm

A problem encountered with any algorithm that needs a priori information is how to get a "good" or "reasonable" initial guess of that information. This is very critical when the algorithm is applied to extremely nonlinear problems because a poor choice of initial conditions can cause the algorithm to converge on an erroneous solution. In addition, a good guess may reduce the number of iterations needed for the solutions to converge.

Target tracking algorithms process data collected from various sensors to generate a tracking solution. It is possible to generate a guess for the initial conditions of the target from these data. Such techniques have been used in satellite orbit determination (Reference 3). The method chosen for use in this study requires inputs of both bearing and frequency measurements from two different sensors. Beginning with the tangent of the observed bearing measurement from sensor i:

$$\tan \hat{\beta}_i = \frac{\sin \hat{\beta}_i}{\cos \hat{\beta}_i} = \frac{y-y_i}{x-x_i}$$

the following linear equation relates the target's coordinates, x and y, to those of the sonobuoys, x_i and y_i,

$$x \sin \hat{\beta}_i - y \cos \hat{\beta}_i = x_i \sin \hat{\beta}_i - y_i \cos \hat{\beta}_i.$$

If a bearing measurement from sensor j is obtained, then the following set of linear equations can be solved for the target's position vector:

$$x \sin \beta_i - y \cos \beta_i = x_i \sin \beta_i - y_i \cos \beta_i$$
$$x \sin \beta_j - y \cos \beta_j = x_j \sin \beta_j - y_i \cos \beta_j$$

Note that accurate position vectors of the sonobuoys are required to solve these equations. If additional bearing data are available from other sonobuoys at this time, they can also be used to generate a least squares estimate of the initial target position vector. When least squares estimation procedures are used, an initial covariance can be computed for this position vector.

A similar procedure is available for determining the initial velocity vector which uses simultaneous bearing and frequency data. The Doppler equation for a non-stationary target and sonobuoy i is

$$f_i = f_o \left(\frac{1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}_i}{\|\bar{r}-\bar{r}_i\| c}}{1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}}{\|\bar{r}-\bar{r}_i\| c}} \right)$$

where

$\bar{r}$, $\bar{v}$ = position and velocity vectors of the target,

$\bar{r}_i$, $\bar{v}_i$ = position and velocity vectors of the buoy i, and

f_o = transmitted target frequency.

By assuming a value for f_o and rearranging the equation, the result is the following linear equation for v

$$\frac{(\bar{r}-\bar{r}_i)}{||\bar{r}-\bar{r}_i||c} \cdot \bar{v} = \frac{f_o}{f_i}\left(1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}_i}{||\bar{r}-\bar{r}_i||c}\right) - 1 .$$

Noting that

$$\frac{\bar{r}-\bar{r}_i}{||\bar{r}-\bar{r}_i||} = \bar{u} ,$$

where u is a unit vector with components cos β_i and sin β_i, the following simple equation results:

$$\bar{u} \cdot \bar{v} = \dot{x} \cos \beta_i + \dot{y} \sin \beta_i$$

$$= c\left[\frac{f_o}{f_i}\left(1 + \frac{\bar{u}\cdot\bar{v}_i}{c}\right) - 1\right] .$$

After receiving another set of simultaneous bearing and frequency data from sonobuoy j, the following set of linear equations can be solved for the target velocity vector

$$\dot{x} \cos \hat{\beta}_i + \dot{y} \sin \hat{\beta}_i = c\left[\frac{f_o}{f_i}\left(1 + \frac{\dot{x}_i \cos\hat{\beta}_i + \dot{y}_i \sin\hat{\beta}_i}{c}\right) - 1\right]$$

$$\dot{x} \cos \hat{\beta}_j + \dot{y} \sin \hat{\beta}_j = c\left[\frac{f_o}{f_j}\left(1 + \frac{\dot{x}_j \cos\hat{\beta}_j + \dot{y}_j \sin\hat{\beta}_j}{c}\right) - 1\right]$$

It is desirable to have several simultaneous data points to insure a good least squares estimate of the velocity and its covariance.

The methods outlined in this section have been implemented and they give adequate estimates of the initial position and velocity of the target. When the solutions are from least squares determinations, it is possible to obtain estimates of the diagonal terms of the state covariance matrix for these parameters. This usually provides a sufficient covariance matrix to be used as a priori input to the target tracking algorithms.

2.3 Constant Tangential and Normal Acceleration Model

Originally, the target motion model used in both the Hybrid and Sequential was a constant Cartesian acceleration model. This model was adequate for both the initialization and tracking phases of both target tracking algorithms, but it was felt that it insufficiently modeled target motions that involved turning maneuvers. Initially, it was proposed to investigate the possibility of adding second-order Taylor series terms to the measurement model to compensate for the trackers' weaknesses in modeling turning maneuvers. However, after more closely examining this problem, it was determined that these turning maneuvers could be better modeled by changing the target's motion model rather than by adding higher order measurement model terms. Consequently, in this study the target's acceleration model was changed from the old Cartesian coordinate model to a first order, Gauss-Markov process for Normal and Tangential acceleration components. This model was found to complicate the system dynamics, but it was also found to have embedded in it, four commonly used motion models --

constant velocity, constant acceleration, constant radius turn and variable radius turn.

The differential equations that describe the tangential and normal accelerations are

$$\ddot{x} = a_T \frac{\dot{x}}{v} - a_N \frac{\dot{y}}{v}$$

$$\ddot{y} = a_T \frac{\dot{y}}{v} + a_N \frac{\dot{x}}{v}$$

where

$$v = \sqrt{\dot{x}^2 + \dots}$$

and a_T and a_N are the constant tangential and normal accelerations. There are four analytical solutions to these differential equations, depending on values of the constants a_T and a_N. These four solutions are:

(1) constant velocity $a_T = 0$, $a_N = 0$

$$\begin{pmatrix} \bar{r}(t + \Delta t) \\ \bar{v}(t + \Delta t) \end{pmatrix} = \begin{pmatrix} I & I\Delta t \\ 0 & I \end{pmatrix} \begin{pmatrix} \bar{r}(t) \\ \bar{v}(t) \end{pmatrix}$$

(2) constant acceleration $a_T \neq 0$, $a_N = 0$

$$\begin{pmatrix} \bar{r}(t + \Delta t) \\ \bar{v}(t + \Delta t) \end{pmatrix} = \begin{pmatrix} I & I(\Delta t + \frac{a_T}{2\|\bar{v}(t)\|} \Delta t^2) \\ 0 & I(1 + \frac{a_T}{\|\bar{v}(t)\|} \Delta t) \end{pmatrix} \begin{pmatrix} \bar{r}(t) \\ \bar{v}(t) \end{pmatrix}$$

(3) constant radius turn $a_T = 0$, $a_N \neq 0$

$$\begin{pmatrix} \bar{r}(t+\Delta t) \\ \bar{v}(t+\Delta t) \end{pmatrix} = \begin{pmatrix} \frac{\|\bar{v}(t)\|}{a_N}\bar{v}(t) & \bar{r}(t) \\ \frac{-a_N}{\|\bar{v}(t)\|}\bar{r}(t) & \bar{v}(t) \end{pmatrix} \begin{pmatrix} \sin\frac{a_N}{\|\bar{v}(t)\|}\Delta t \\ \cos\frac{a_N}{\|\bar{v}(t)\|}\Delta t \end{pmatrix}$$

(4) variable radius turn $a_T \neq 0$, $a_N \neq 0$

$$\begin{pmatrix} \bar{r}(t+\Delta t) \\ \bar{v}(t+\Delta t) \end{pmatrix} = \begin{pmatrix} (1+\frac{a_T}{\|\bar{v}(t)\|}\Delta t)^2 & \left(\frac{\|\bar{v}(t)\|}{a_N}\bar{v}(t) - 2\frac{a_T}{a_N}\bar{r}(t) \quad \bar{v}(t)\right) \\ (1+\frac{a_T}{\|\bar{v}(t)\|}\Delta t) & \left(2\frac{a_T}{a_N}\bar{v}(t) - \frac{4a_T^2 + a_N^2}{a_N\|\bar{v}(t)\|}\bar{r}(t) \quad \bar{v}(t)\right) \end{pmatrix} \cdot$$

$$\begin{pmatrix} \sin\frac{a_N}{a_T}\ln(1+\frac{a_T}{\|\bar{v}(t)\|}\Delta t) \\ \cos\frac{a_N}{a_T}\ln(1+\frac{a_T}{\|\bar{v}(t)\|}\Delta t) \end{pmatrix}$$

Note the complexity of the solutions, especially the variable radius turn. In order to use these equations in a "batch" tracking algorithm, it would be necessary to have statistical tests for model selection and for data interval selection. These are the same four motion models used in the Maximum Likelihood Estimator (MLE) developed at Tracor (Reference 4). From the test results documented in Reference 1, the MLE is not as fast as the Hybrid or Sequential because it takes considerable computer time to select the proper motion

model and data interval. To avoid these problems, the motion model and data interval selection features have been dropped in both the Hybrid and Sequential.

For the Sequential, data interval and motion model selections could be dropped because an EKF is used to generate target state estimates. This EKF modifies the estimates for a_T and a_N and rectifies the target's state vector estimates for each measurement processed. These estimates are valid only for this update point and do not need to be saved beyond the next update point. This process of not saving past estimates relaxes some of the restrictions on this tracker. The Hybrid, on the other hand, does use a "batch" filter to initialize the tracker. When batch processors are used to generate estimates, one is more troubled with the data interval and motion model selection because all the measurements and tracker estimates are mapped back to the initial epoch of the trajectory. If the wrong motion model or data interval are chosen, the tracker cannot successfully map all the information back to this initial epoch. However, the Hybrid only uses this "batch" filter to initialize the tracker and then switches to an EKF as soon as adequate initial conditions have been found. Typically, only 50 to 100 seconds of target data are needed to successfully converge onto a set of initial conditions. Within this time frame, one rarely finds that a submarine will initiate some maneuver which would require a motion model change. Furthermore, it is believed that enough flexibility has been built into the Hybrid's motion model to compensate partially for a single maneuver. Since the "batch" filter is used only to initialize the tracker and since no drastic change in the target's motion is expected over the relatively short initialization phase, the motion model and data interval selection features of the MLE have not been incorporated into the Hybrid.

Reviewing the four common motions embedded in the normal-tangential acceleration model, one notes the complexity of the analytical solutions, particularly for the variable radius turn. Complicating these equations even more is the way the acceleration directions are coupled to the velocity components, leaving a system of coupled differential equations. Furthermore, to analytically solve these differential equations, one would have to use a model selection feature (which we are seeking to avoid) to determine which analytical solution to use. Faced with all these problems, it was decided that the way to implement the new motion model was to numerically solve the differential equations with a classical, fourth order Runga-Kutta algorithm described in Reference 5. This allows the differential equations to be integrated without performing motion model tests and without decoupling the equations of integration.

2.4 Sensor Motion Model

Previously, the sonobuoys were assumed to be stationary. Realistically, they drift due to ocean currents and surface winds. Under these circumstances, it is possible for each sonobuoy to drift in different directions. Furthermore, mobile platforms are often used to gather acoustic data. It is important to model the sensor motion as accurately as possible in order to successfully model the measurement process. To maintain generality, it was assumed that associated with every data point is an estimate of the position and velocity vectors of the sonobuoy. If the position vector was required at an intermediate time, the following constant velocity model is used:

$$\bar{r}_i(t+\Delta t) = \bar{r}_i(t) + \bar{v}_i(t)\Delta t$$

This is an adequate model when small Δt's are used. An advantage to using this model, which has been incorporated in the trackers, is that both algorithms can now process data from mobile sensors, such as towed arrays and hull-mounted systems as well as data from drifting sonobuoys. Furthermore, Doppler shifts can be better estimated by the tracking algorithms' measurement model because sensor motion is accounted for in the estimates.

2.5 New Data Models

Besides modifying the Hybrid's and Sequential's target and sensor motion models, four new measurement types were added to the measurement models. These new measurement types include one active measurement and three passive, two-sensor data types. The additions of these measurement types enable the Hybrid to process most of the data types available from acoustic signal processors.

The active range measurement for sensor i is defined as:

$$\rho = \frac{1}{2} c \Delta t$$

where

$$\rho_i = \| \bar{r} - \bar{r}_i \|$$

c = speed of sound in water

Δt = time interval between transmitting and receiving the reflected signal.

Time difference of arrival of a signal between two sensors has also been added to the measurement model. This measurement is modeled as:

$$\tau_{ij} = \frac{\rho_i - \rho_j}{c}$$

Another two-sensor measurement that can now be used is the Doppler ratio. This data type is modeled as:

$$\frac{f_i}{f_j} = \left(\frac{1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}_i}{\|\bar{r}-\bar{r}_i\|\,c}}{1 + \frac{(\bar{r}-\bar{r})\cdot\bar{v}}{\|\bar{r}-\bar{r}_i\|\,c}}\right)\cdot\left(\frac{1 + \frac{(\bar{r}-\bar{r}_j)\cdot\bar{v}}{\|\bar{r}-\bar{r}_j\|\,c}}{1 + \frac{(\bar{r}-\bar{r}_j)\cdot\bar{v}_j}{\|\bar{r}-\bar{r}_j\|\,c}}\right)$$

Finally, the last intersensor data measurement that was modeled, the Doppler difference, is defined as:

$$f_i - f_j = f_o\left(\frac{1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}_i}{\|\bar{r}-\bar{r}_j\|\,c}}{1 + \frac{(\bar{r}-\bar{r}_i)\cdot\bar{v}}{\|\bar{r}-\bar{r}_i\|\,c}}\right) - f_o\left(\frac{1 + \frac{(\bar{r}-\bar{r}_j)\cdot\bar{v}}{\|\bar{r}-\bar{r}_j\|\,c}}{1 + \frac{(\bar{r}-\bar{r}_j)\cdot v_i}{\|\bar{r}-\bar{r}_j\|\,c}}\right)$$

2.6 Test Evaluation Criteria

The primary parameter used to qualitatively analyze the performance of the tracking algorithms is the

position error. In order to avoid confusion in terminology, position error is also called distance error. Previously, when the position error was used for analysis of tracking results, three measures of performance were computed. The first measure was the average position error over the entire track. The next measure was the convergence time, defined as the time required for the tracker to converge to a steady state error value smaller than some specified value. The last performance parameter was the predicted position error incurred by projecting the tracker estimates forward for 300 seconds after the last measurement was received. When the average and predicted distance errors were below 500 meters and when the position error converged to a steady state value below 500 meters, then the tracking algorithm performance was deemed good. The same basic parameters were used to measure tracking performance for this study, but two of the definitions have been changed.

In this study, the average position error is measured only after convergence has been achieved. Before, convergence was attained when the position error reached a steady state value below 500 meters. Now convergence results when the position error reaches any steady state. Rather than compute values for convergence time and distance errors directly, they are now obtained from the plots of the position error versus time. The predicted distance error is still directly computed.

In addition to the plot of the position error, plots of the tangential and normal components of the position error were developed as an analysis aid. The tangential, or along track, direction is defined to be along the velocity

vector and the normal, or across track, direction is perpendicular to it. Given the definition

$$\delta\bar{r} = \bar{r}_E - \bar{r}$$

where $\bar{r}$ and $\bar{r}_E$ are the true and estimated position vectors, respectively, the tangential or along track error is:

$$\delta\bar{r}_T = \delta r \left(\frac{\bar{v}}{\|\bar{v}\|} \right),$$

$$\text{where } \delta r = \| \delta\bar{r} \| .$$

The normal or across track error is

$$\delta\bar{r}_N = \delta r \left(\frac{\bar{w}}{\|\bar{w}\|} \right),$$

where $\bar{w}$ is the vector normal to the velocity vector. By looking at these plots, one is able to make qualitative conclusions about the geometrical effects of target tracking.

Another qualitative analysis aid is the plot of the true and estimated trajectories. Included on these plots are the sonobuoy positions and the 300 second prediction point. Combined, these four plots give an analyst the tools needed to evaluate tracking algorithms.

2.7 Test Scenarios

The test sets used in testing the Hybrid and Sequential algorithms were the revised OCCD case 1 and case 8 described in Appendix A, Table A.IV. These two scenarios are referred to as scenarios 1 and 2, respectively, in this subsection. In both, the target moved through the tri-tac pattern of stationary sonobuoys at a constant speed of 5 meters per second on a straight line course. For Scenario 1, the buoy separation distance was 8000 meters, the signal-to-noise ratio at one yard from the target was 82 dB and the signal integration time was 20 seconds; while in Scenario 2, these quantities were 5000 meters, 70 dB and 5.0 seconds, respectively. The scenarios were not comprehensive because they did not contain maneuvering targets or new data types. The data used were bearing and frequency measurements from a simulated DIFAR processor. Despite the shortcomings of these scenarios, they were adequate for providing a preliminary appraisal of the target tracking capabilities of the modified Hybrid and Sequential algorithms.

2.8 Test Results

Tests were conducted on the Hybrid and Sequential to compare the new algorithms' performances to the old versions. In the tests, no initial conditions were given to the algorithms except for an *a priori* state covariance matrix that was required by the old Sequential. All other algorithmic inputs were the same.

Tables 2.I and 2.II contain the quantitative results for all of the trackers' solutions for scenarios 1 and 2. These tables contain the three tracking performance factors

TABLE 2.I

SCENARIO 1 TRACKING RESULTS

ALGORITHM	CONVERGENCE TIME (secs)	AVERAGE POSITION ERROR (meters)	PREDICTED POSITION ERROR (meters)
New Hybrid	110	25	100
Old Hybrid	130	30	102
New Sequential	110	25	101
Old Sequential	130	30	104

TABLE 2.II

SCENARIO 2 TRACKING RESULTS

ALGORITHM	CONVERGENCE TIME (secs)	AVERAGE POSITION ERROR (meters)	PREDICTED POSITION ERROR (meters)
New Hybrid	100	60	647
Old Hybrid	100	100	677
New Sequential	130	60	647
Old Sequential	N/A	N/A	N/A

for the four tracking algorithms studied, the old and new versions of both the Hybrid and the Sequential. Figures 2.1 through 2.14 display the performance of the tracking filters for both scenarios. The plots display the true trajectory with a solid line and the estimated trajectory with a dashed line. The x and y axes are the x and y components of the trajectories in meters. The position error curves display the RMS distance error between the estimated and true trajectories as functions of time. These tables and figures are used below to evaluate the performance of each of the algorithms.

Results from Table 2.I for Scenario 1 indicate that all four algorithms' performances were comparable. None of the performance measures showed any significant difference to indicate a particular algorithm's superiority, but the new algorithms did converge sooner than the older ones. The trajectory plots of Figures 2.1 through 2.4 show that each algorithm could estimate a rather smooth trajectory and that the predicted position was not far from the true position. Overall, the distance error curves in Figures 2.5 through 2.8 indicate that all of the tracking algorithms' estimated tracks were fairly close to the true ones. Since the distance error plots for all the algorithms were small for this scenario, the along-track and across-track distance error curves have not been included. This scenario was very favorable in terms of signal-to-noise ratio and tracking geometry, so good tracking performance was expected and found for all of these algorithms.

The less favorable Scenario 2 produced poorer results. The old Sequential algorithm was unable to initialize in this case and therefore was unable to track the target. Since the new Sequential algorithm did track this same target, the modifications such as the inclusion of the initial guess

algorithm and the new SKF initializer proved to be beneficial in making the new version superior to the older one. From Table 2.II it is seen that the quantitative tracking results were nearly the same for the other three trackers. The overall results were good, but the large prediction errors above 500 meters were unacceptable. Looking at Figures 2.9 through 2.11, the algorithms show significant divergence between the predicted track and the true one over the last portion of these plots. The old Hybrid algorithm had more trouble than the others early in the track, but the difficulty disappeared as the scenario progressed. The two new algorithms had less difficulty because of their ability to generate good initial estimates from the data, even with poor sonobuoy coverage. The tracking relied principally on two sonobuoys early in the track with most of the data coming from No. 1 and some from No. 3. Very little or no data came from No. 2 during the first 400 seconds. The position error plots of Figure 2.12 through 2.14 show that after convergence, there was a jump in the steady state position error, particularly in the newer algorithms. This increase was a result of the data beginning to enter the algorithms from sonobuoy No. 2, while sonobuoy No. 3 was beginning to loose contact with the target. Further investigation indicated that a pronounced jump in the along-track position error occurred when sonobuoy No. 2 began detecting the target. Since sonobuoy No. 2 is nearly along the track, the change in steady state value can be attributed to poor data accumulated by it.

2.9 Conclusions and Recommendations

The modifications made to the Hybrid and Sequential tracking algorithms were designed to improve their performances and expand their range of applications. Tests conducted on the algorithms did not examine the effectiveness

of the new data types because no adequate data generation was available. The performance of the algorithms with active range, time-difference of arrival, Doppler ratio and Doppler difference data must be evaluated at a later date. Since all simulated data came from stationary buoys, the sonobuoy motion model remains to be evaluated.

Modifications that were tested include the initial guess algorithm, the new Sequential initializer and the new acceleration model. The quantitative results showed that all the algorithms were comparable in tracking performance. However, qualitatively, the results indicated that the new versions of the Hybrid and Sequential algorithms are slightly superior to their former versions. In the case of the Sequential, the overall analysis shows that the new version is superior to the original because it was able to track the targets from both scenarios.

To fully evaluate the Hybrid and Sequential, comprehensive tests should be devised to exercise every modification made. These tests should have scenarios that use both maneuvering and non-maneuvering targets, and also include various combinations of data types such as bearing, frequency and range or bearing and time difference of arrival. Any new test should implement non-stationary sensors, such as drifting sonobuoys or towed arrays.

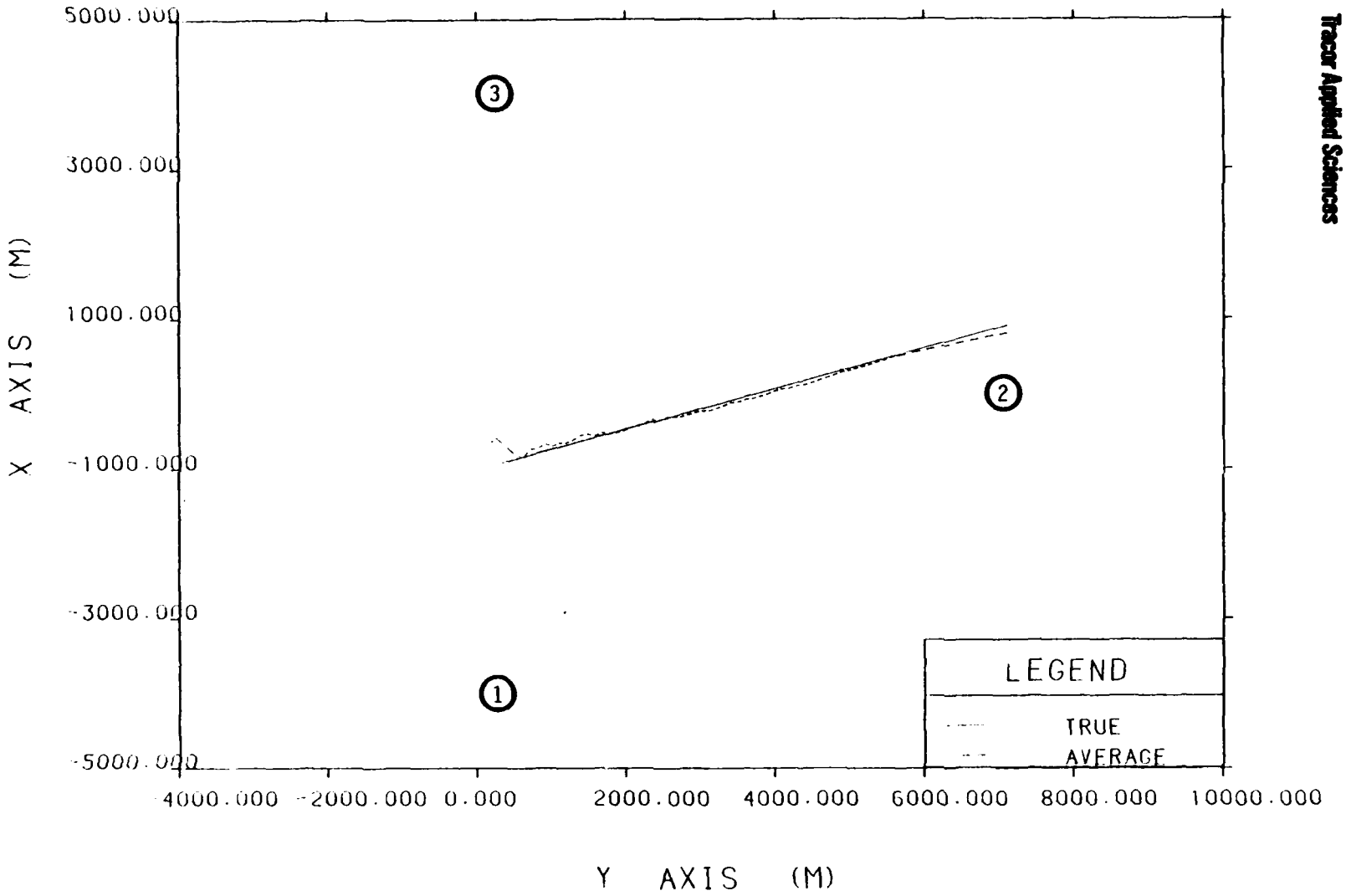


Figure 2.1 - NEW HYBRID TRAJECTORY (SCENARIO 1)

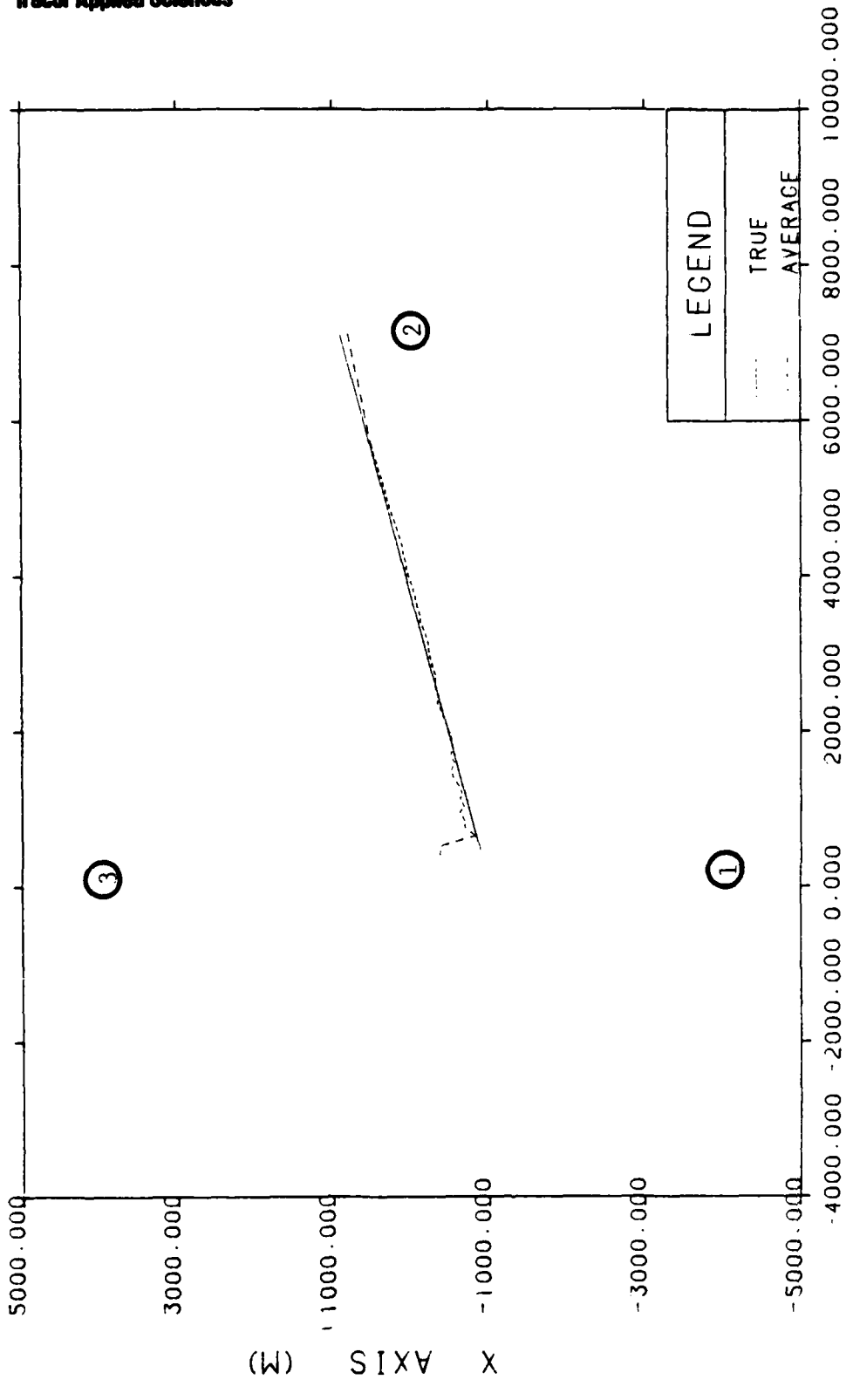


Figure 2.2 - OLD HYBRID TRAJECTORY (SCENARIO 1)

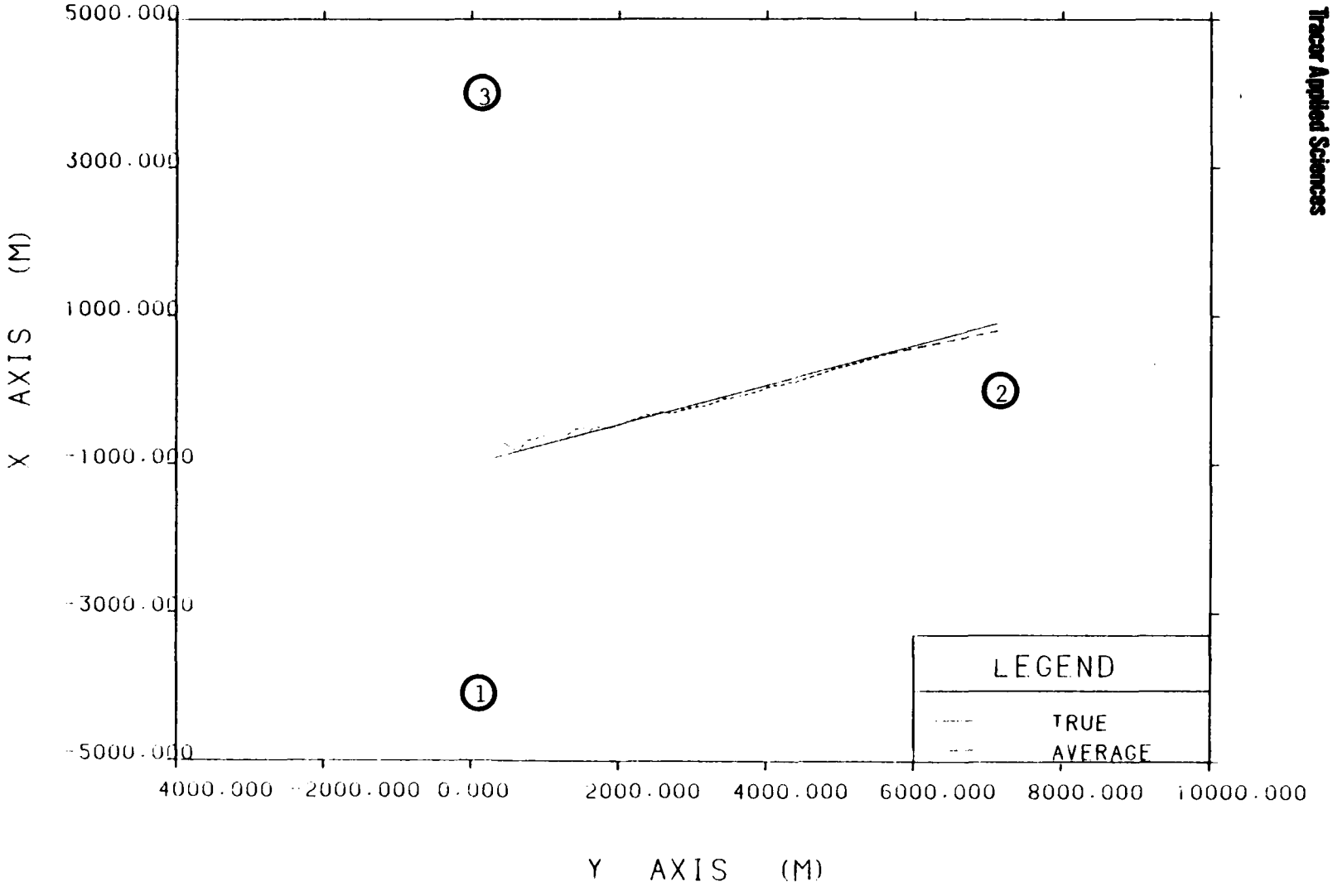


Figure 2.3 - NEW SEQUENTIAL TRAJECTORY (SCENARIO 1)

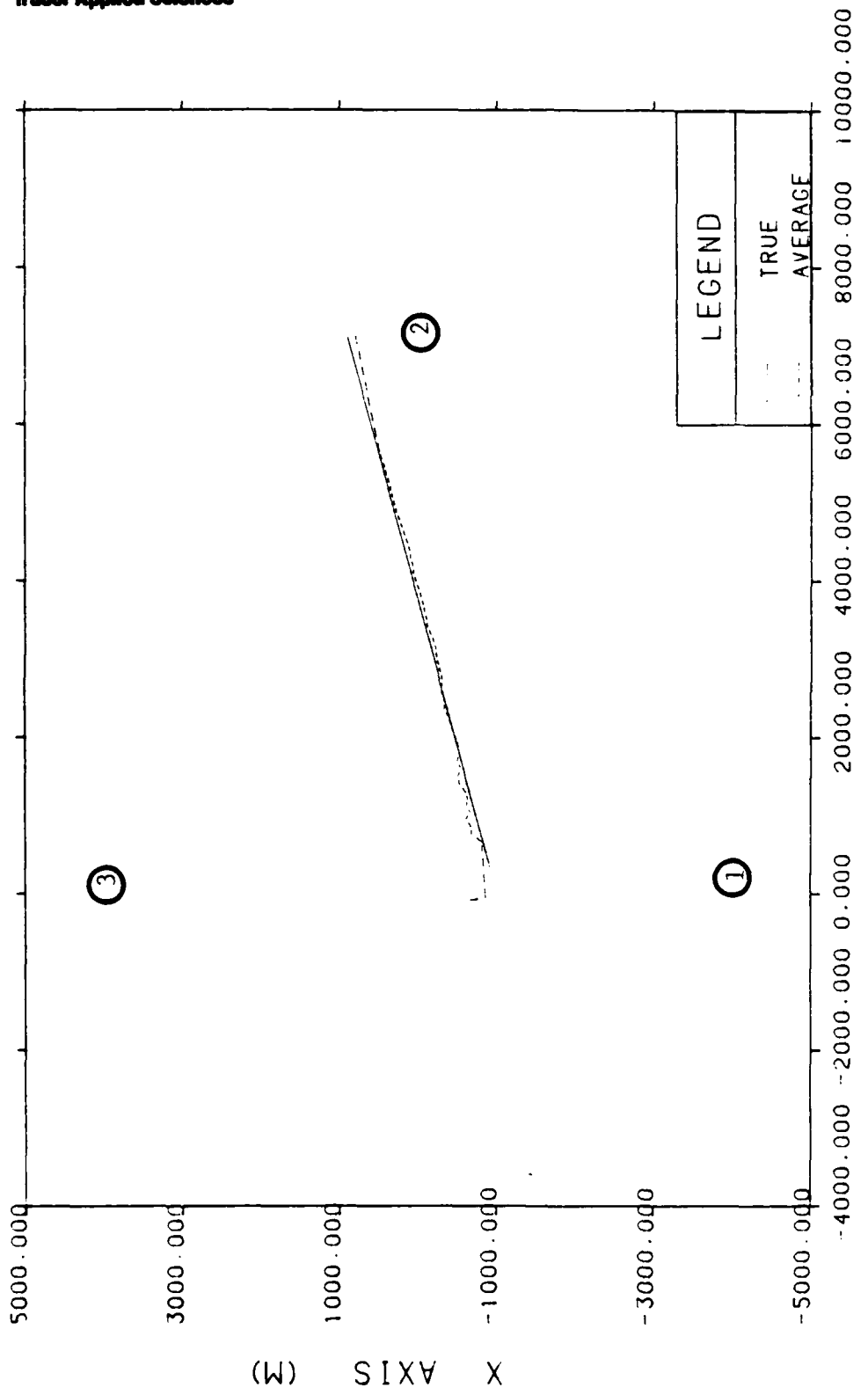


Figure 2.4 - OLD SEQUENTIAL TRAJECTORY (SCENARIO 1)

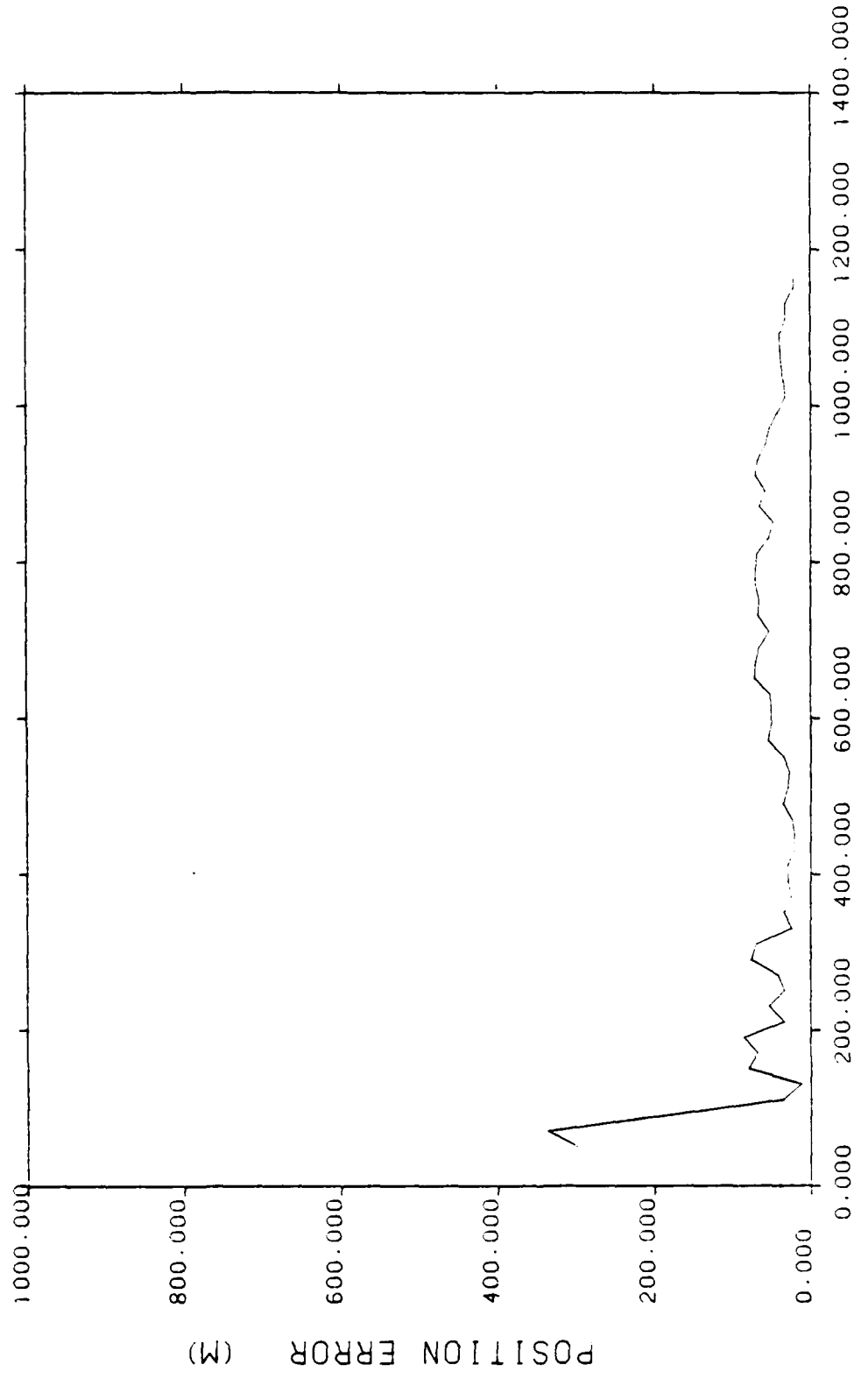


Figure 2.5 - NEW HYBRID POSITION ERROR (SCENARIO 1)

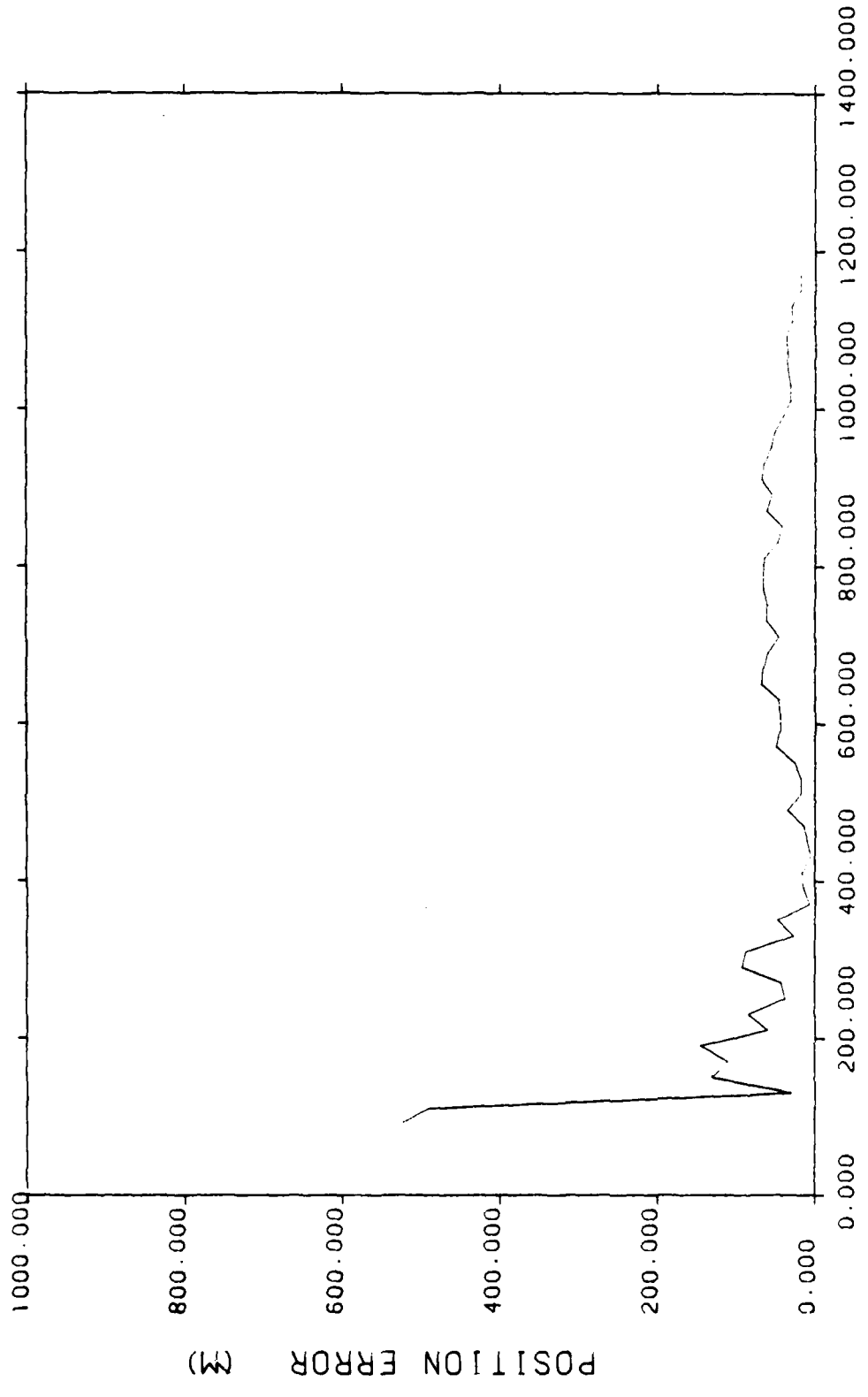


Figure 2.6 - OLD HYBRID POSITION ERROR (SCENARIO 1)

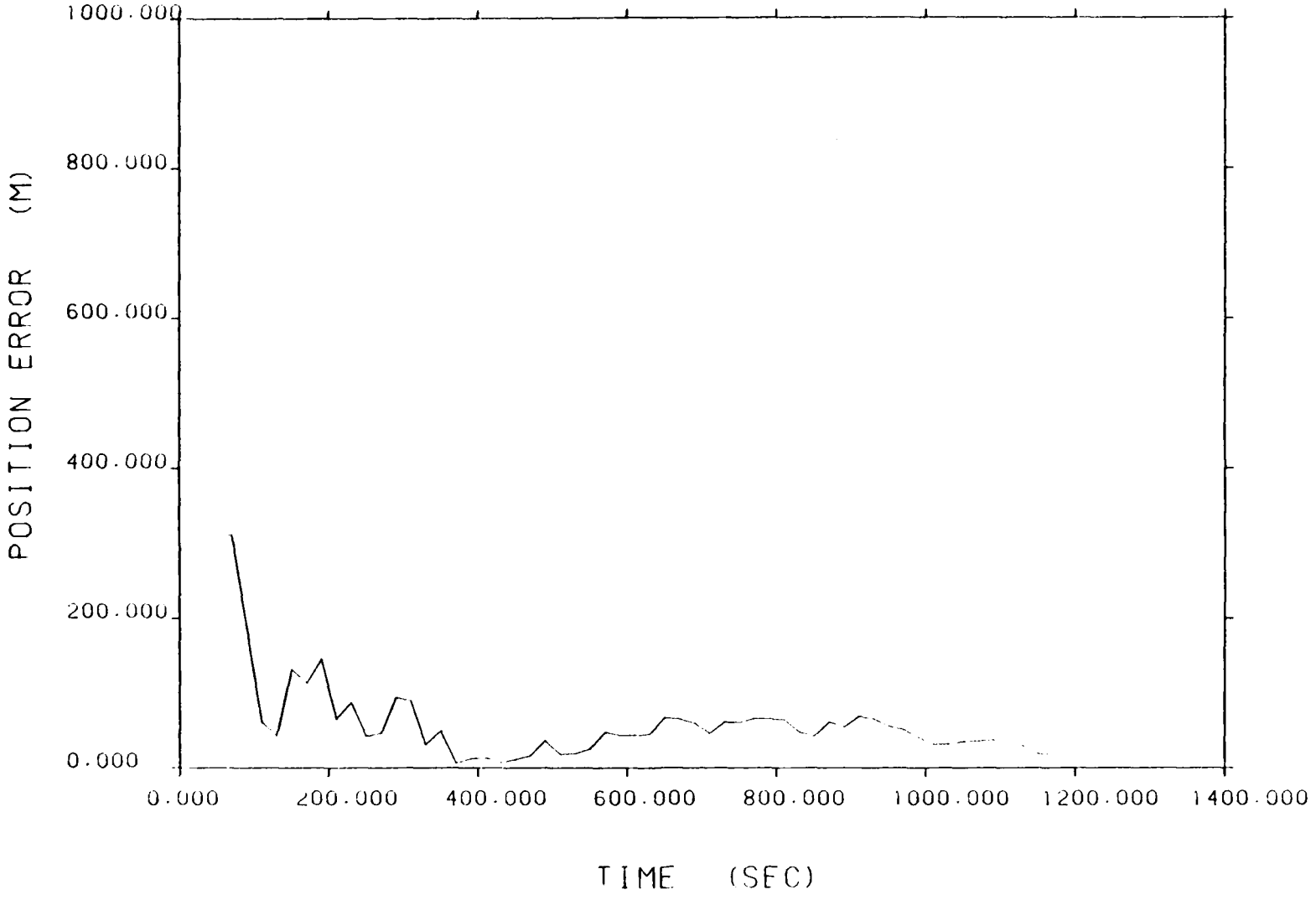


Figure 2.7 - NEW SEQUENTIAL POSITION ERROR (SCENARIO 1)

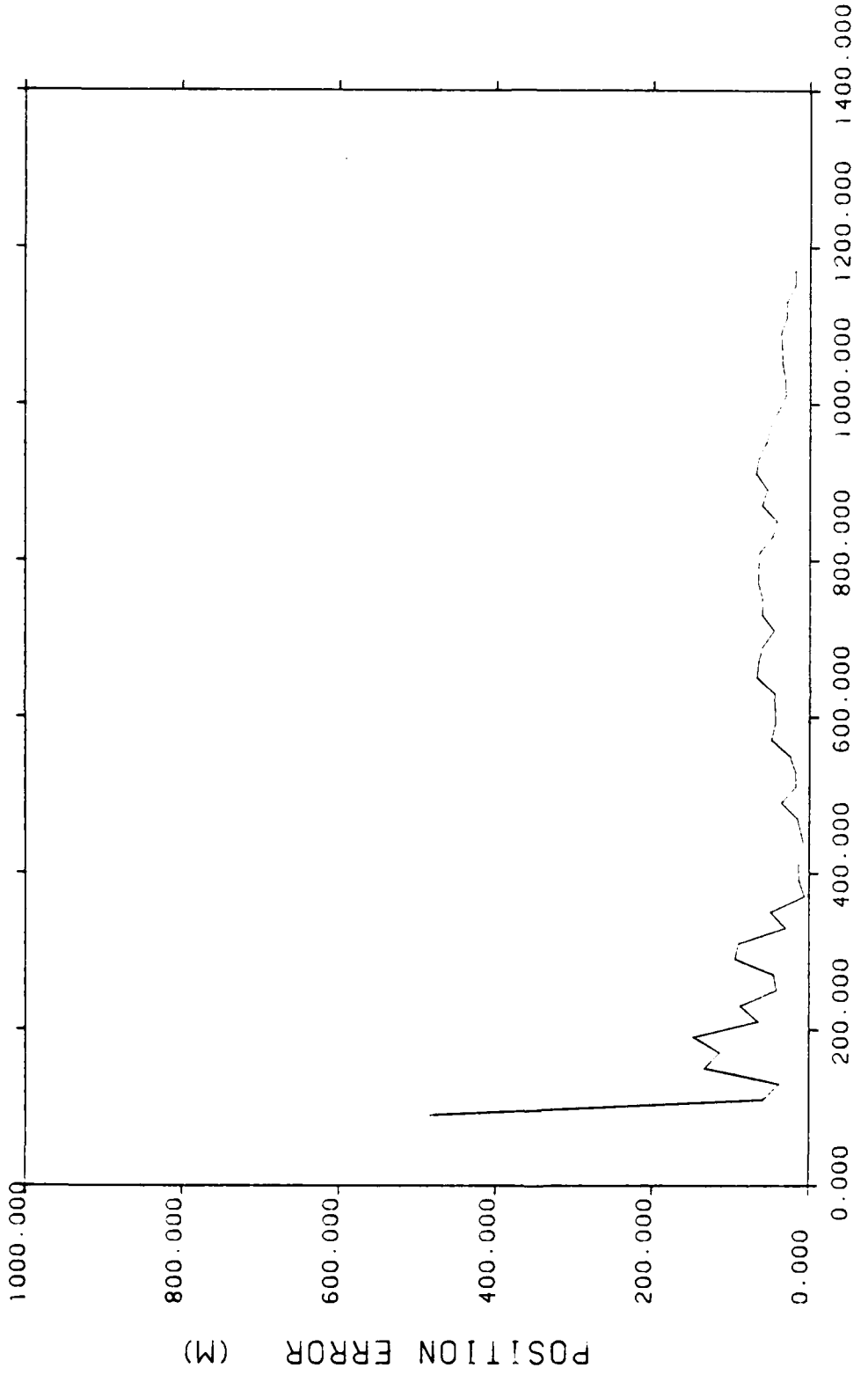


Figure 2.8 - OLD SEQUENTIAL POSITION ERROR (SCENARIO 1)

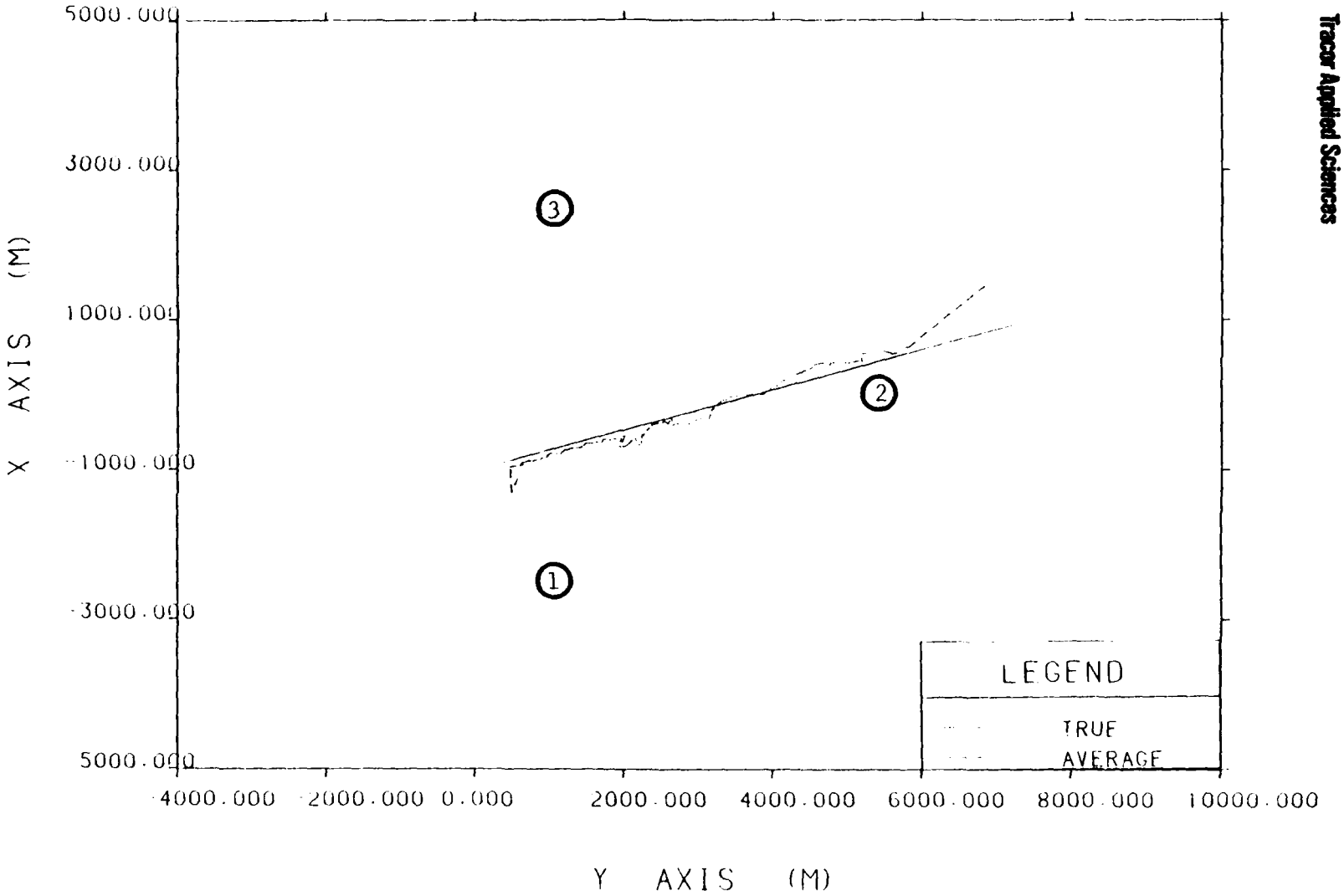


Figure 2.9 - NEW HYBRID TRAJECTORY (SCENARIO 2)

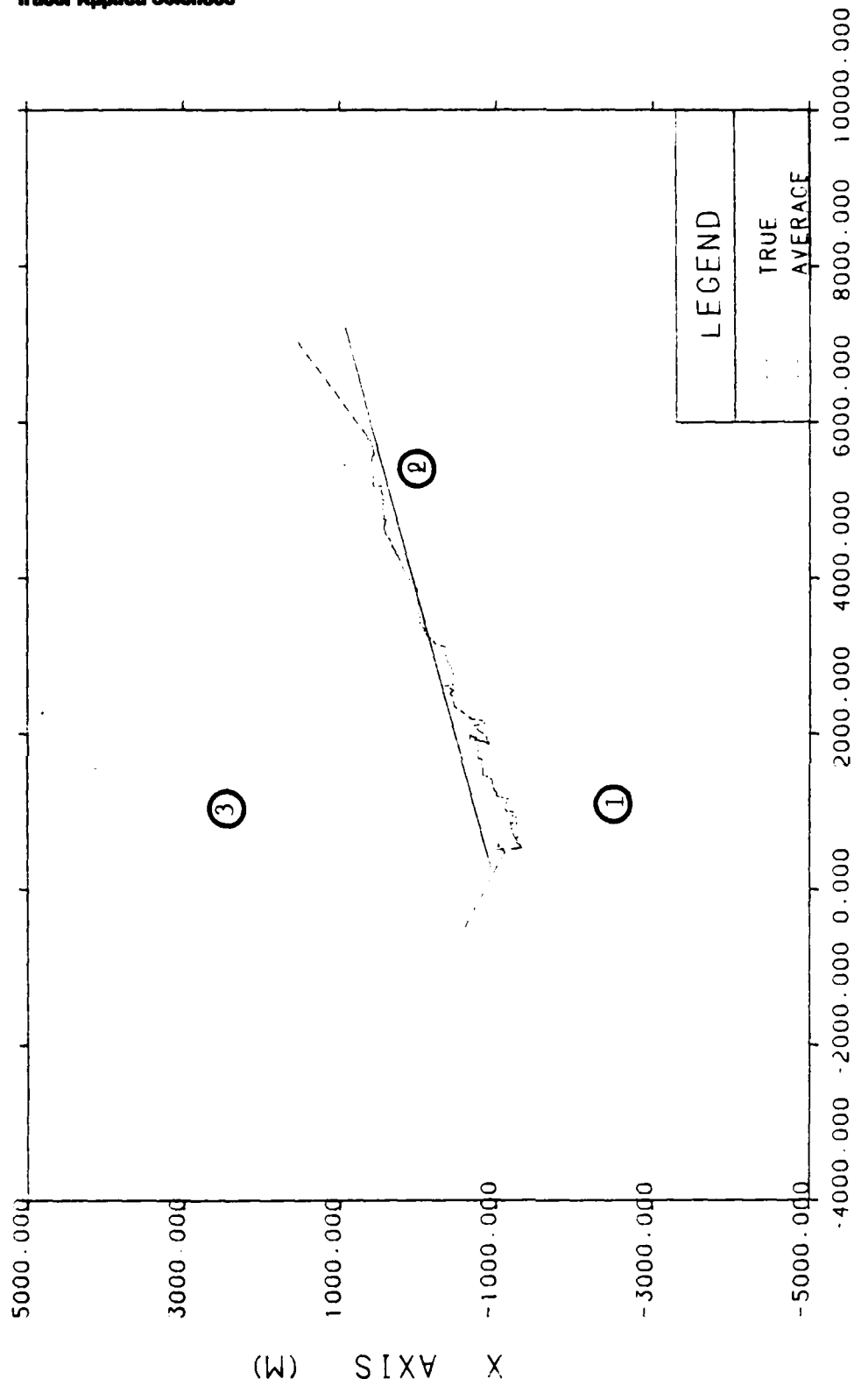


Figure 2.10 - OLD HYBRID TRAJECTORY (SCENARIO 2)

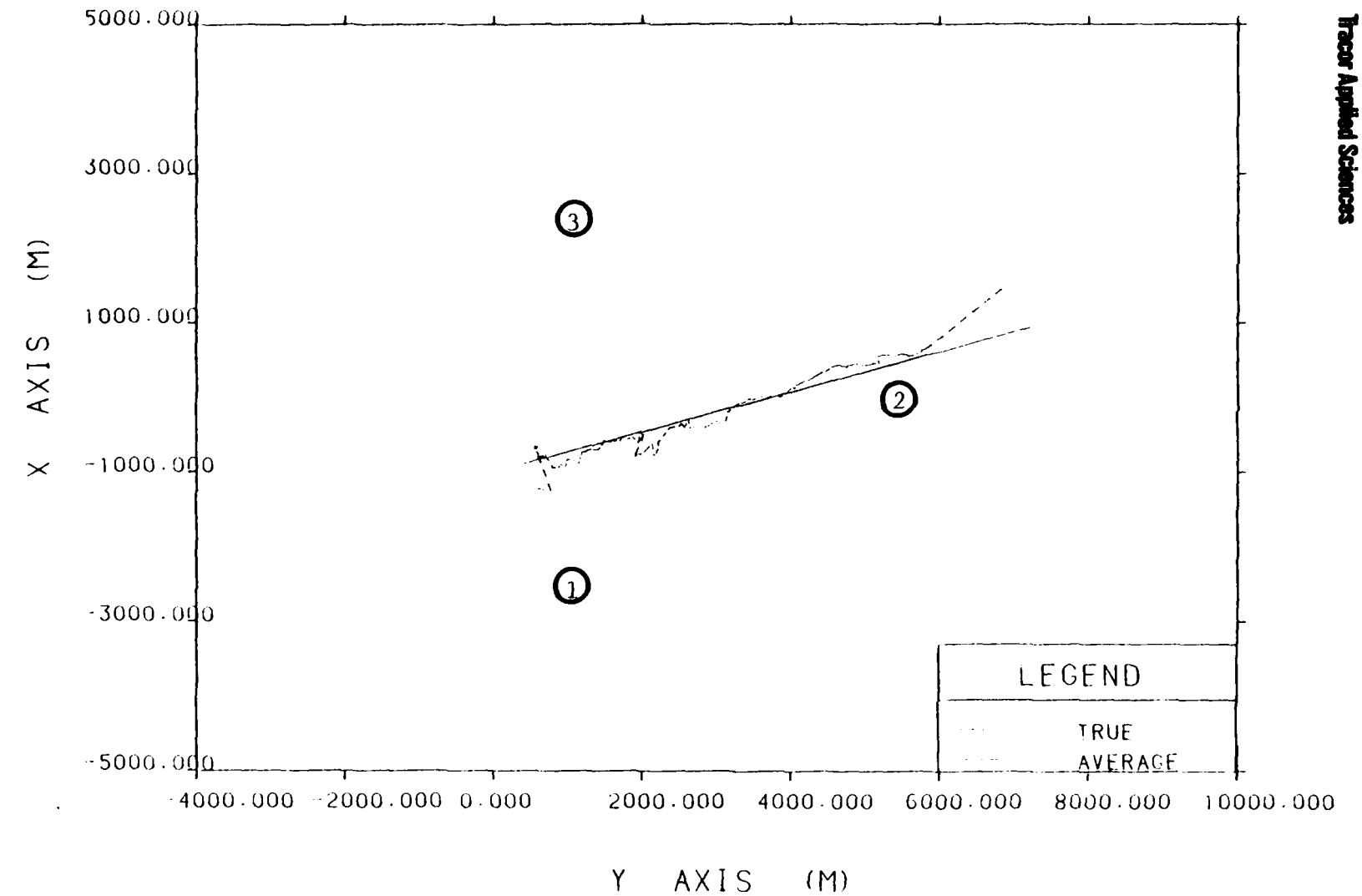


Figure 2.11 - NEW SEQUENTIAL TRAJECTORY (SCENARIO 2)

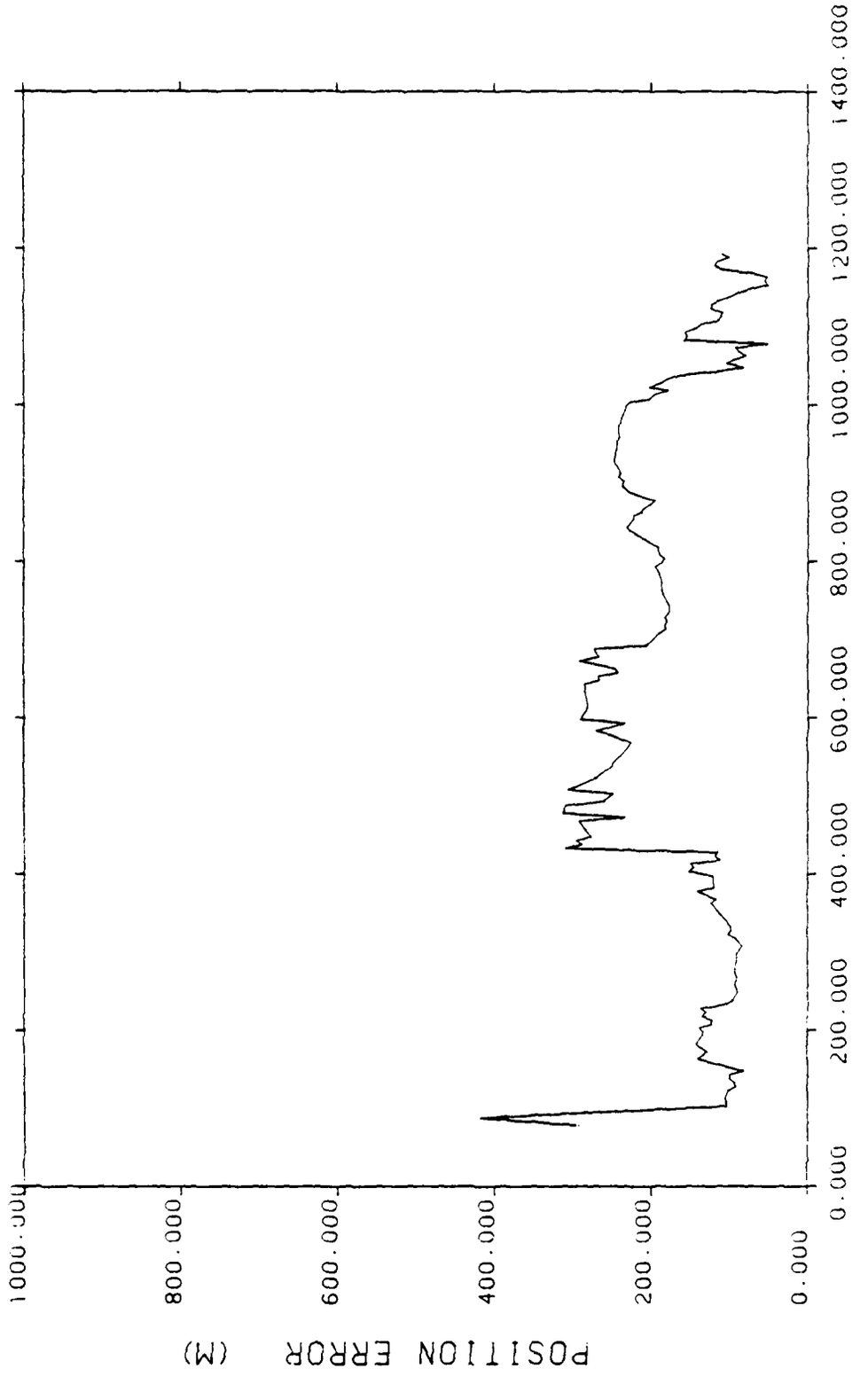


Figure 2.12 - NEW HYBRID POSITION ERROR (SCENARIO 2)

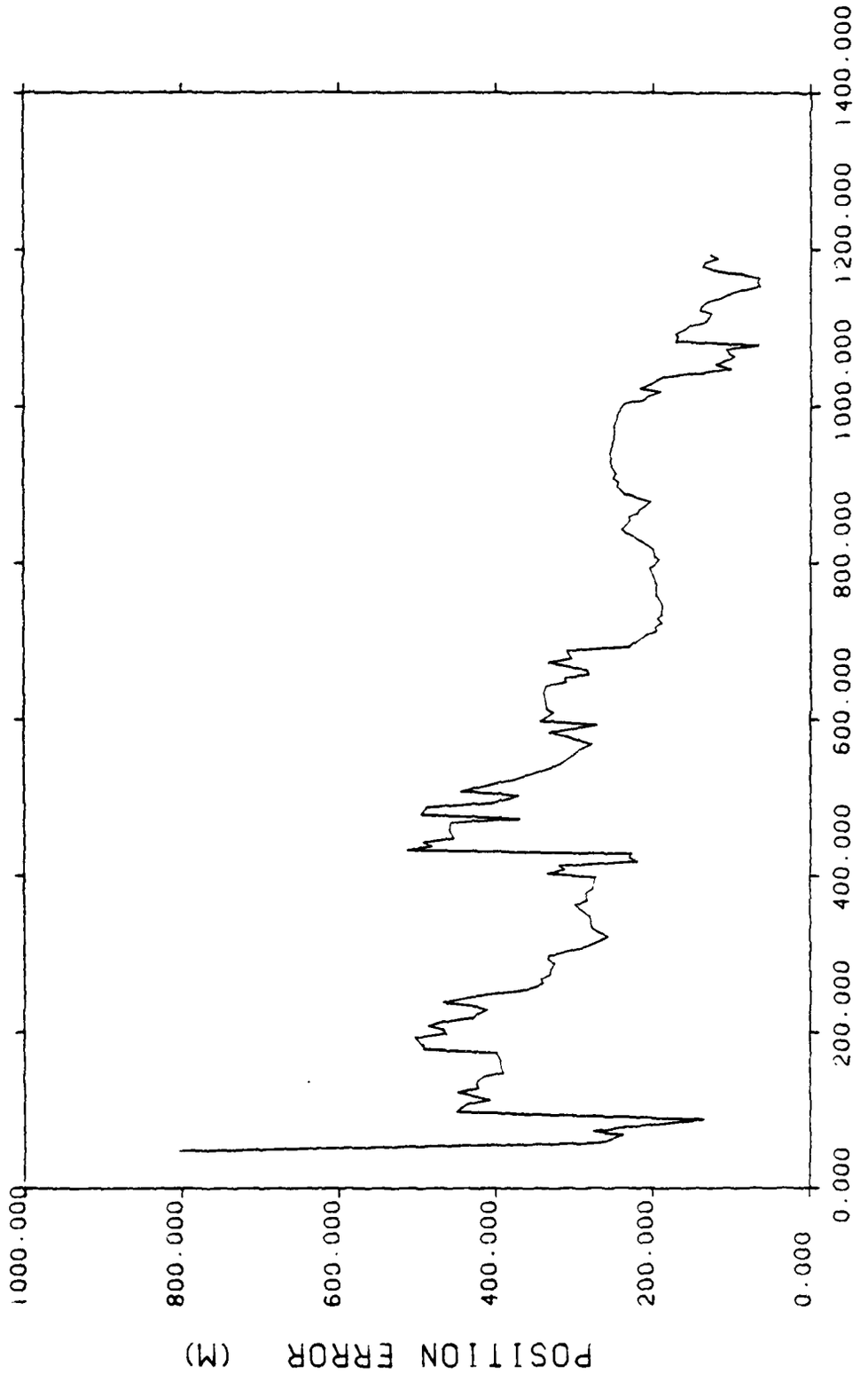


Figure 2.13 - OLD HYBRID POSITION ERROR (SCENARIO 2)

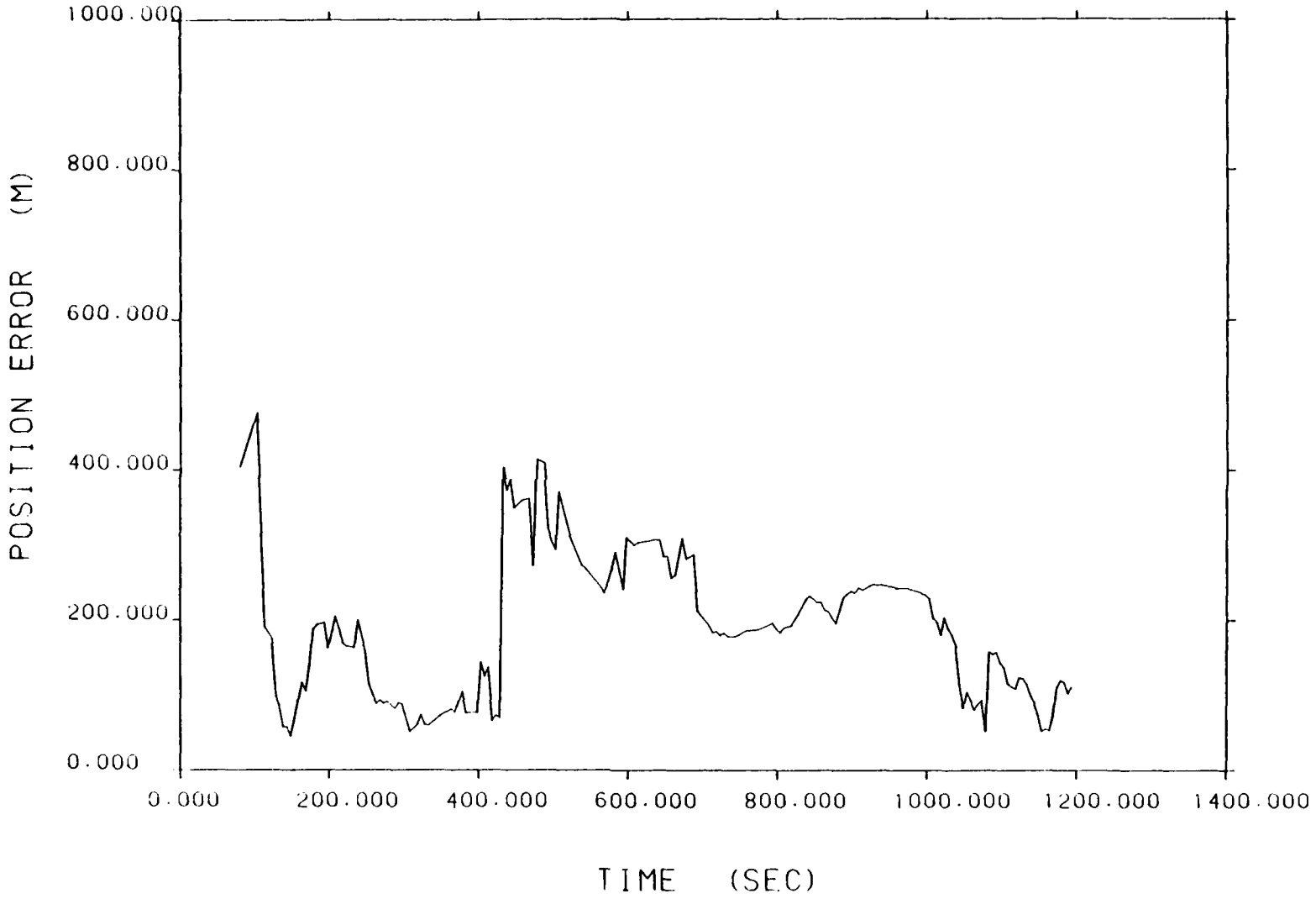


Figure 2.14 - NEW SEQUENTIAL POSITION ERROR (SCENARIO 2)

3.0 SIMULATION OF SINGLE TARGET DATA

Many types of acoustic processors are available for generating measurement estimates such as frequency and bearing for target tracking. Passive systems include OMNI sonobuoys which are used to make narrowband frequency estimates and DIFAR sonobuoys which are used to estimate both narrowband frequency and bearing measurements. Another class of passive detection systems include hull mounted sensors and towed arrays. These systems are used by both surface ships and submarines. Hull mounted and towed array systems are capable of making long range, narrowband frequency and bearing estimates as well as broadband bearing estimates. Besides the passive systems mentioned, active systems are used that transmit high energy pulses and listen for return signals to make measurement estimates. Active systems use return times to generate range estimates for a target. Some of these systems also generate Doppler frequency and bearing estimates based on the return signal. However, the algorithms considered in this study primarily require only passive narrowband acoustic measurments for inputs. Furthermore, it was felt a priori that narrowband frequency and bearing estimates for a given target would be sufficient for use in sorting data for the multiple target problem. Therefore, this study concentrates only on data processors that generate passive, narrowband frequency and bearing estimates for the single target and multiple target tracking problems.

3.1 Passive Narrowband Frequency and Bearing Simulator

A computer program was developed which generates simulated narrowband frequency and bearing estimates (Reference 1). The simulated data from this model yield

non-Gaussian measurement errors that are fairly reasonable when compared to samples of true sea data.

The data generation program models a square law detector which uses a MAX-OR processor to compute frequency estimates and an arctangent processor to compute bearing estimates. A schematic of the simulator model is given in Figure 3.1. This figure shows a sensor that receives signal plus ambient noise and that is followed by a comb filter bank. This comb filter bank consists of a fixed number of frequency bins that are Δf Hz wide. The value Δf is chosen by the user so that the frequency estimate's resolution is controllable. Following the comb filter bank, a square law detector is used to detect the level of omnidirectional power present in each frequency bin. The noise spectrum is assumed normalized. The integration time for the square law detector is inversely proportional to the bin width, Δf, set for the comb filter bank. From this inverse relationship, one can see that when long integration times are used, fine frequency bin widths will result for the comb filter bank. A post detection integrator follows the square law detector in the data simulation program. This post detection integrator allows the processor to average the output of the square law detector over a fixed number of samples to reduce the variance of the estimates. This averaging process increases the probability that the MAX-OR processor will pick the signal peak of the spectrum and reject random noise peaks. The post detection integrator was not used in this study because only single sample outputs from the square law detector were used to generate measurement estimates. After the optional post detection integrator, a MAX-OR processor is used to analyze the power levels in each bin of the comb filter bank. From the single bin chosen by the MAX-OR processor, frequency and bearing estimates are

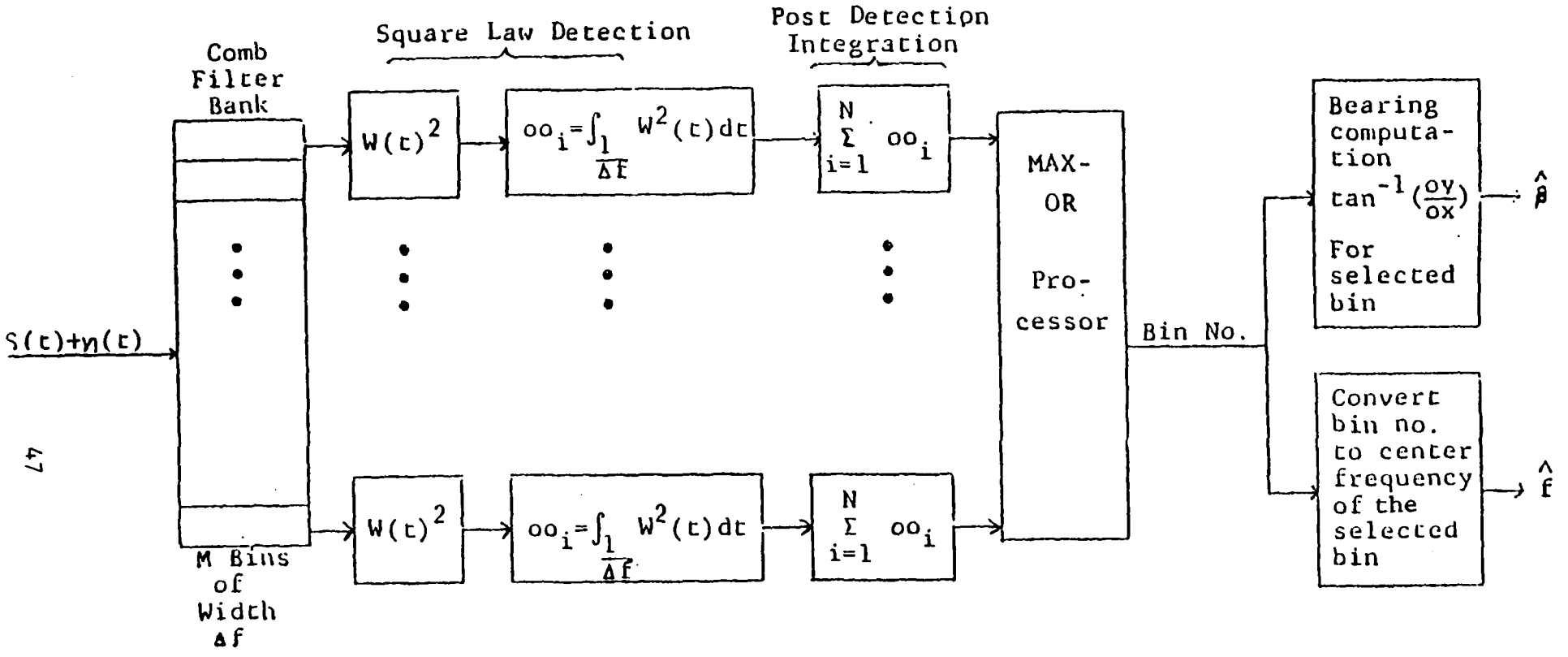


Figure 3.1 - General Processor For DIFAR Buoy

produced. To determine the frequency estimate, the bin number of the chosen frequency bin is converted into a frequency value which corresponds to the center of that bin. The simulated X-channel and Y-channel outputs from the two dipole sensors for the chosen frequency bin are then analyzed by an arctangent processor to produce the bearing estimate. If the omni-directional power level for the chosen frequency bin exceeds a specific fixed value, the measurements are accepted; otherwise they are rejected. This threshold level is usually set to limit severely the number of false alarms than can be accepted.

Another major portion of the data generation process is concerned with scenario kinematics and time variables. It is necessary to compute a variety of time varying parameters for a simulated scenario before the processor model described above can be used to generate a sequence of frequency and bearing estimates for each sensor involved in the simulation. Positions and velocities of all the participants are passed to the data generation program so that bearings, ranges, SNR's, and Doppler shifts can be computed at each time increment for each sensor in a specified scenario. A program which accepts sensor positions, target initial conditions, and subsequent target motions as inputs has been developed to provide the required functions for the processor model. The following subsection details some of the considerations involved in computing the processor's computed SNR in this model. Details of the kinematic and geometry portions of the scenario generation model will not be discussed here.

3.2 SNR Calculations for the Output Measurements

Signal-to-noise ratio (SNR) calculations figure prominently in the computing and the weighting of the frequency

and bearing estimates. The accuracy of the calculations for both frequency and bearing estimates are affected by the SNR of the detected signal. Furthermore, the threshold test, which determines whether or not an estimate is accepted, is based on the SNR of the power in the omnidirectional channel for the chosen frequency bin. Because of the importance of the SNR computed for the detected signal in the data simulation program, great care is taken to model most of the factors which affect the SNR detected by the MAX-OR processor.

Representative values for the target's radiated signal strength and for the ambient noise level are chosen. The strength of the target's signal is chosen to conform with values for various classes of targets. The ambient noise level is chosen to model the effects of surface, marine life, and distant surface ship noise. Given the target's signal strength and the ambient noise level, the SNR in dB at one yard from the target in a 1 Hz band is the difference between these two levels.

Two different factors are then considered in modeling the degradation of the signal's SNR found when the signal is transmitted through the water to the sensor (Reference 6). One loss is called the attenuation loss. This loss is a function of the radiated frequency and the range or distance from the target to the sensor. Attenuation loss is much greater for high frequencies and is almost negligible for the low frequencies used in this study. A more important loss encountered with acoustic signals is the spreading loss. For the ranges associated with deployed, narrowband systems, the spreading loss is approximated by a simple 20 log R loss in dB, where R is the magnitude of the distance in yards from the target to the sensor. This one-way propagation model assumes

spherical spreading in an isovelocity medium. While this is not strictly true in the open ocean, it does account for the main signal loss for close range sonobuoy operations. Variations in sound velocity will cause the actual losses to be more or less than the modeled values; however, the general study results from using the 20 log R loss should be indicative of typical ocean results. If the results for particular environmental conditions are required, tabulated propagation losses may be substituted for the simple model. These two losses, attenuation and spreading, are modeled in the data simulation program to compute a reasonable SNR value for the signal detected by a sonobuoy's receiver.

Besides the propagation losses, other factors are modeled and affect the SNR value computed for the detected signal. Both the ambient noise level and the signal strength level are scaled at each time step by random noise terms to model random fluctuations in these two values. These random contributions affect the computed SNR for the omnidirectional channel for each bin and create more realistic fluctuations for each bin.

The other factor considered in computing the SNR is the possible smearing of one narrowband tone over several bins during a given integration period. Due to the changing dynamics and geometries of a target moving relative to a particular sonobuoy, the Doppler shifted frequency of the received signal varies with time. Particularly during CPA, i.e., when the Doppler shift changes from compression to expansion, the narrowband tone will slide through several frequency bins both above and below the unshifted frequency value. It is quite possible that the detected signal can slide through two or more frequency bins within one integration time

when the Doppler shifted frequency value changes in time. When this happens, the detected omnidirectional power for one signal is effectively split over two or more frequency bins. Because the MAX-OR processor picks only the single bin with the most power and ignores all adjacent bins, the omnidirectional power in the chosen bin will really contain only a fraction of the signal's total power during that integration interval. This causes a noticeable drop in the SNR for the detected signal for this integration interval. This can lead to an apparent fading or even a loss of the signal. This smearing of the signal over several bins is modeled in the data generation program. The program samples the signal many times over one given integration period, and places simulated omnidirectional powers in the appropriate frequency bin for each sample. At the end of the integration period, a percentage is computed for the amount of time the signal spends in each frequency bin. The percentage for each bin then multiplies the power in that bin to produce a simulated power distribution for that integration period.

Propagation losses, random fluctuations in the mean target signal level and the mean ambient noise level, and the possible smearing of the signal across several frequency bins are considered in the data simulation model. These effects are considered to be the major factors which affect the SNR value detected by the MAX-OR processor. Using these models for the SNR calculations, data is produced that contains periods of signal fading and signal loss. Furthermore, the data is non-Gaussian and effectively tests the data sorting capabilities and target tracking capabilities of candidate algorithms.

3.3 Simulated Measurement Error Curves

After developing this data generation simulator, error standard deviation curves for both the bearing and frequency estimates were needed to weight these measurements for tracking purposes. A sample of over 22,000 data points was used to compute the means and standard deviations of the measurement errors for many different SNR values. Generally, over 1,000 samples were generated for each SNR range to assure statistical accuracy in the calculations of the means and standard deviations of the measurement errors. To produce the sample, one fixed sonobuoy and a single, non-moving target were used to generate simulated data. The range between the sensor and the target remained fixed at 5,000 meters for all of the data gathered. To obtain measurements over the full range of SNR values, the input ambient noise level and target strength level were varied from run to run. A frequency cell size of 0.1 Hz was used to analyze the frequency spectra. After the simulation runs were made, the data were merged into one large data set. The data were then sorted into ranges of SNR values, and the error statistics were determined for each of these ranges. In this fashion, the mean and the standard deviation of the errors for the simulated frequency and bearing measurements could be determined as functions of the SNR computed for the detected signal.

The mean of the errors for both the frequency and bearing estimates were near zero for all values of SNR. The standard deviations of the measurement errors for both the frequency and bearing estimates were found to be quite large for low SNR's and to approach zero very quickly for medium to high SNR's. These statistics confirm that poor measurement estimates are made for weakly detected signals but more

accurate measurements are estimated for more strongly detected signals. The curves for the standard deviations of the frequency and bearing errors as functions of the SNR are shown in Figures 3.2 and 3.3, respectively. With these curves, realistic threshold levels for this simulator can be chosen. Furthermore, accurate weights for the simulated frequency and bearing measurements can be computed as a function of the SNR computed for the detected signal.

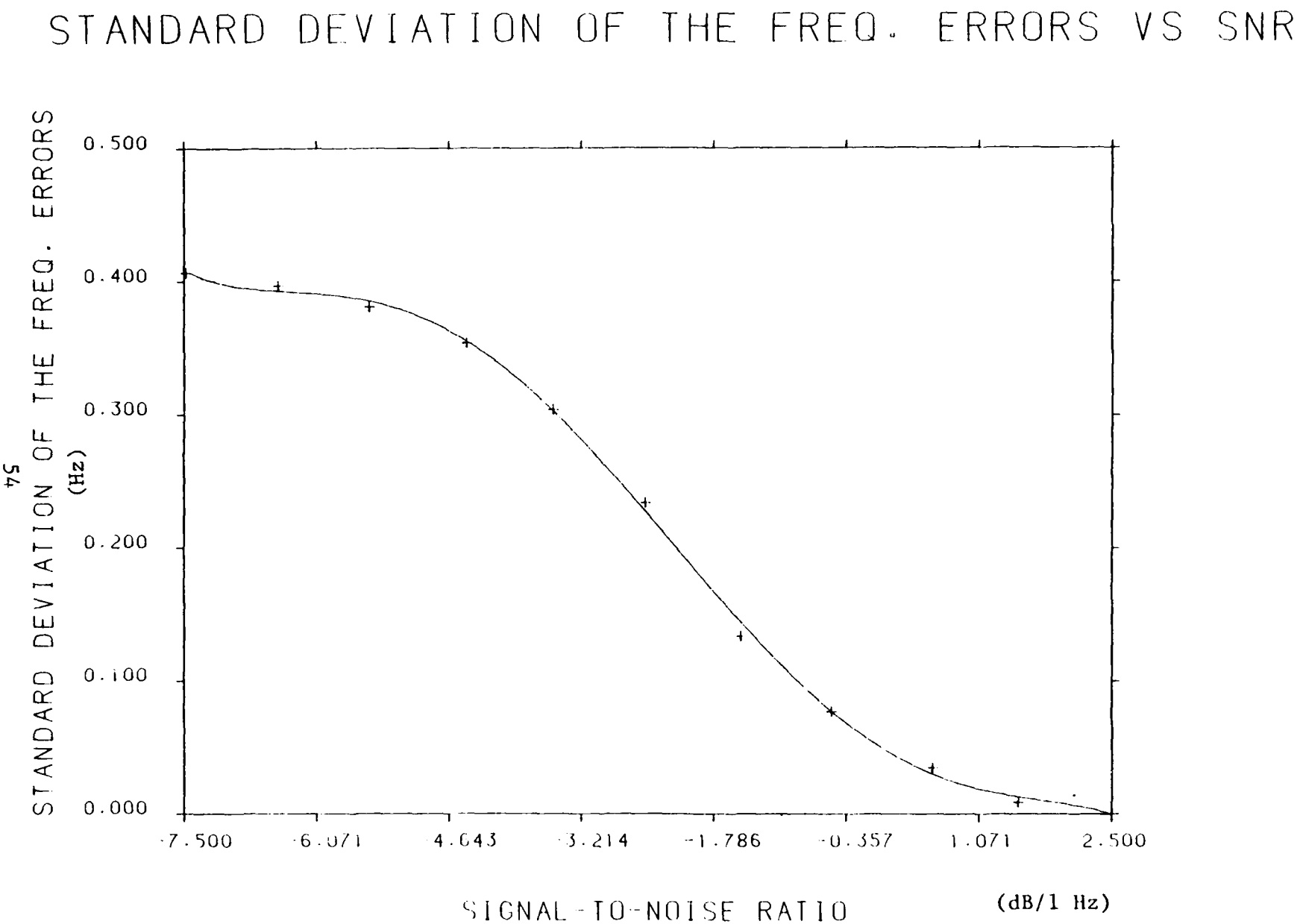


Figure 3.2 - Standard Deviation Of The Simulated Frequency Errors vs. SNR

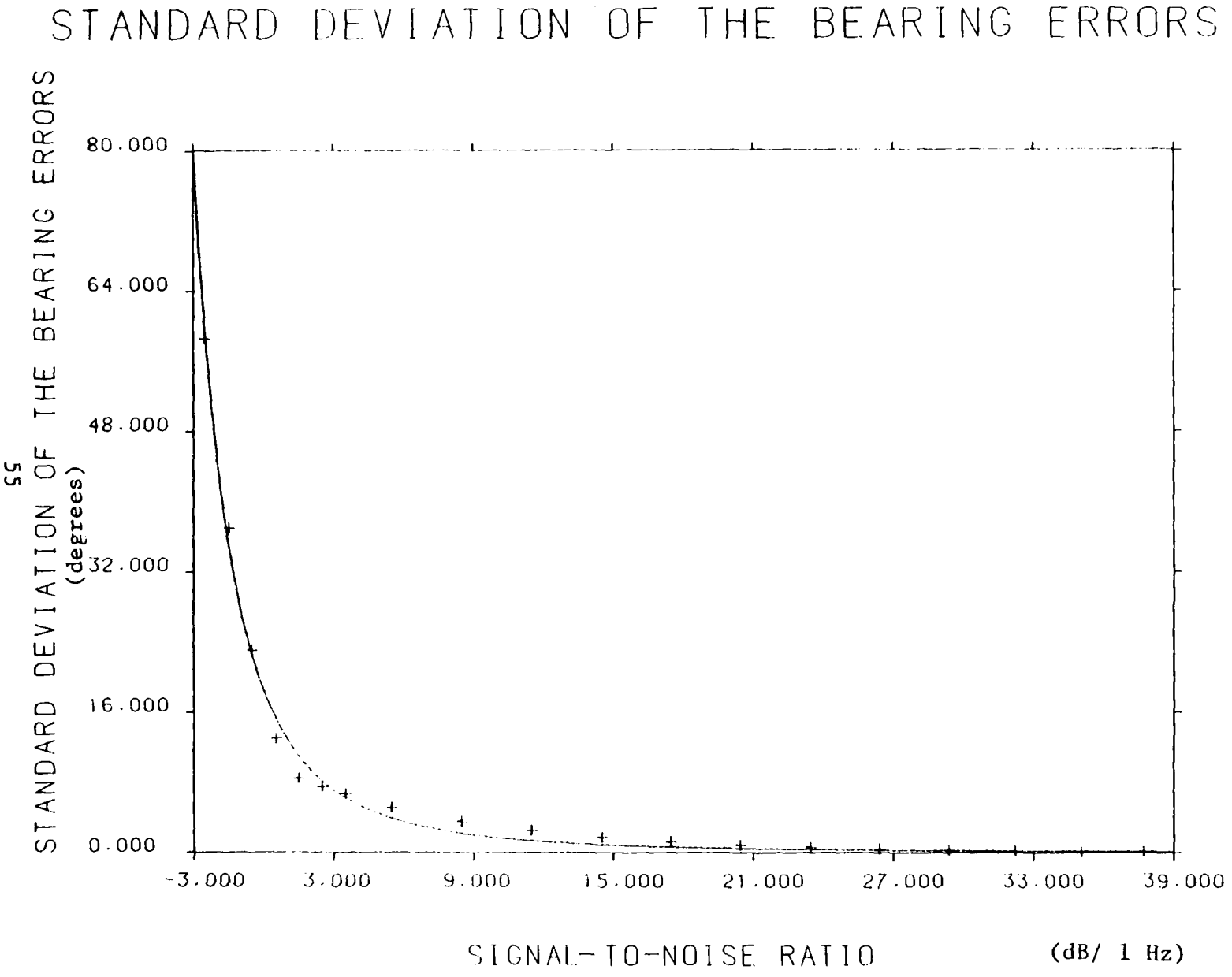


Figure 3.3 - Standard Deviation Of The Simulated Bearing Errors vs. SNR

4.0 SIMULATION OF MULTIPLE TARGET DATA

Detailed descriptions of the two multiple target scenarios are presented below. This section describes these scenarios used in the investigation of this problem as well as the two techniques used to generate the simulated multiple target data. With these data, the preliminary investigation into a proposed data sorting approach was made possible.

4.1 Multiple Target Scenarios

Two scenarios were chosen to be used for the initial multiple target study. The single target scenario generation program, described in Section 3.0, was used to simulate the motion for each separate target used in the scenarios. This scenario generation program contained models that allowed an analyst to simulate constant velocity trajectories or to simulate maneuvering trajectories that utilize either velocity changes or course heading changes. Initially, however, only constant velocity, constant heading trajectories have been used to reduce the number of variables in the study.

4.1.1 Scenario One - The first scenario consisted of three different targets which were observed by a tri-tac sono-buoy pattern. This scenario is shown in Figure 4.1 and described by the information in Table 4.I. Each target started at a different location with a different speed and course heading. All targets maintained their original course and speed. The total simulated scenario lasted for 200 seconds for all three targets. Measurements were updated at 10 second intervals. This scenario was chosen to determine how well our

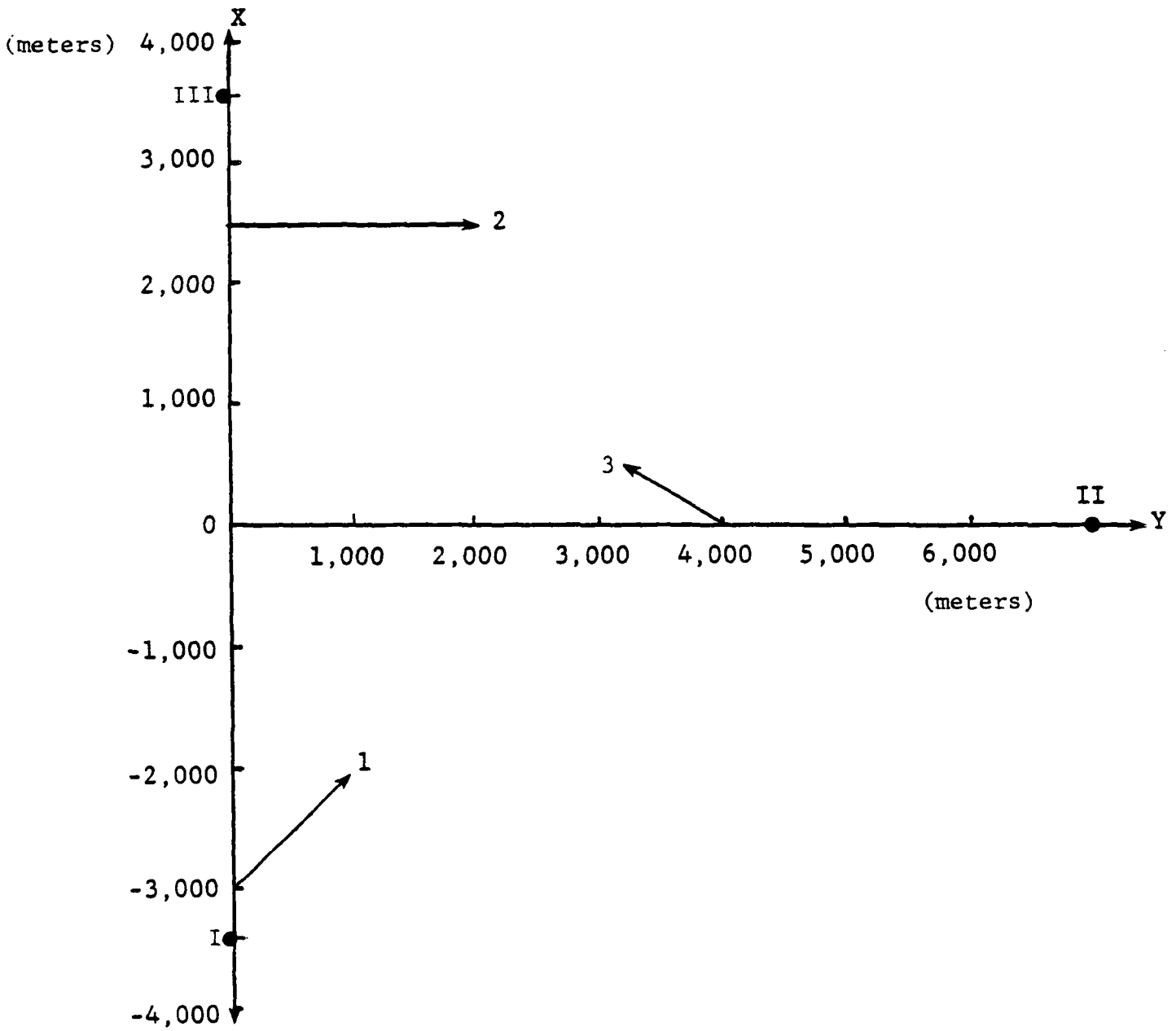


Figure 4.1 Scenario 1, 3 Targets

TABLE 4.I

DESCRIPTION OF SCENARIO 1

Buoy Information

Sensor	X (m)	Y (m)	V (m/sec)
I	-3,500	0	0
II	0	7,062	0
III	3,500	0	0

Target Information

Target	X_o (m)	Y_o (m)	V (m/sec)	θ (°)	X_f (m)	Y_f (m)
1	-3,000	0	6	45	-2,151	849
2	2,500	0	9	90	2,500	1,800
3	0	4,000	4	300	400	3,307

All Targets: f_0 = 150 Hz

data sorting approach could separate data from targets with very different dynamics and geometries but that remained within the observation range of the tri-tac sonobuoy pattern.

4.1.2 Scenario Two - A second, more difficult scenario was designed to test the limitations of the data sorting program. This scenario is shown in Figure 4.2 and is described in detail in Table 4.II. Two targets traveled at precisely equal velocities along parallel paths that were separated by 1500 m. These two trajectories ran for 400 seconds. The course headings for both targets perpendicularly intersected an imaginary line which joined sensors 1 and 3 of the tri-tac pattern. This scenario was chosen to generate data that would create problems for sensors 1 and 3. Since the two targets traveled parallel trajectories, very little difference in bearing estimates for the two targets could be detected by sensors 1 and 3. Furthermore, if both targets transmitted narrowband tones at the same or very nearly the same center frequency, little or no difference would be detected in the Doppler shifted frequencies received by sensors 1 and 3. By studying this scenario, it could be determined how similar two different signals could be before the data sorting program fails to separate the two target data into correct individual data sets for each target.

4.2 Multiple Linetracker Data

The first multiple target data simulation scheme employed the previously described DIFAR data generation program to create linetracker data for each of the targets in the scenarios. The results were then merged into a set of multiple target data for each sonobuoy. The data for each target were created as though an individual linetracker was dedicated to

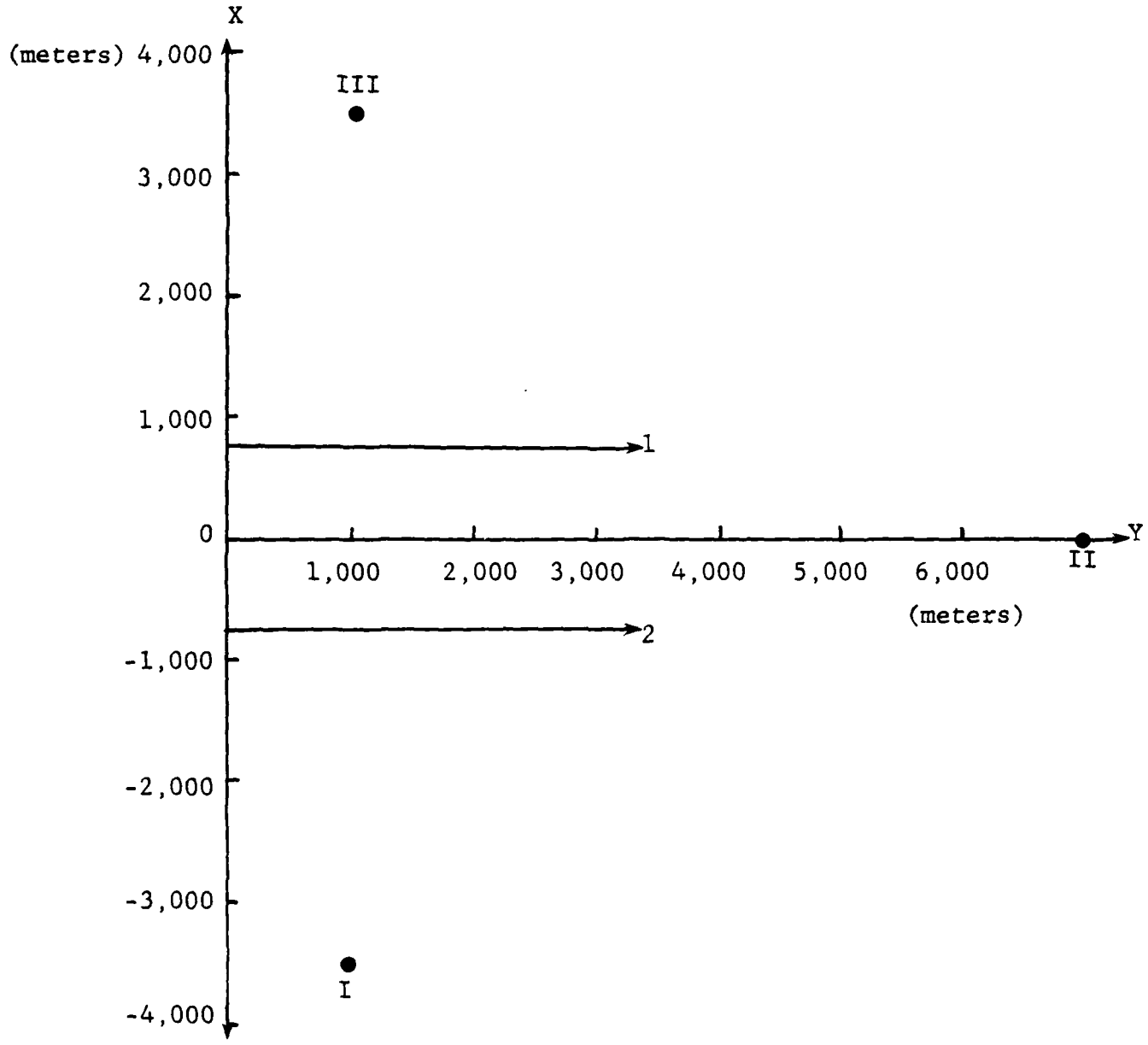


Figure 4.2 Scenario 2, 2 Targets

TABLE 4.II

DESCRIPTION OF SCENARIO 2

Buoy Information

Sensor	X (m)	Y (m)	V (m/sec)
I	-3,500	1,000	0
II	0	7,062	0
III	3,500	1,000	0

Target Information

Target	X_o (m)	Y_o (m)	V (m/sec)	θ (°)	X_f (m)	Y_f (m)
1	750	0	8	90	750	3,200
2	-750	0	8	90	750	3,200

$$f_{o_{T1}} = 150 \text{ Hz}$$

$$[f_{o_{T1}} - f_{0_{T2}}] = 0.0,\ 0.1,\ 0.2,\ 0.3,\ 0.4, \text{ and } 0.5 \text{ Hz}$$

that target with no outside interference from any other source. This assumption is not always valid, but it was used in these simulations. Separate sets of linetracker frequency and bearing measurements were created for each target in the scenario. Then the data merging program merged the data by the time tag and observing buoy number to create a single set of multiple target data for each sensor. This last step destroys line identification information that would be provided if individual line trackers were actually used to track the separate target lines. In a sense, this step makes the data more realistic. This merged data could actually be produced if the MAX-OR processor in the DIFAR simulator was replaced with a processor that thresholds and then picks the n (n = 2, 3, 4, etc.) largest peaks instead of only the single largest peak at each output time. In any event, the data described in this subsection will be referred to as multiple linetracker data in the remainder of this report. Table 4.III contains a sample set of the merged linetracker data for all three targets as simulated for sensor I of scenario 1. Multiple linetracker data for all three sensors in both scenarios were generated in this fashion.

4.3 Simulated Multiple Target Frequency Spectra

As noted above, practical questions were encountered with simulating the multiple linetracker data as described above. It was assumed that no interference from the other signals was encountered by a linetracker that was set to observe a specific frequency line. Sometimes narrowband tones are so closely clumped together that the fixed width of the linetracker's observation window makes it impossible to isolate one line from all of the others. Particularly when the MAX-OR processor is used to pick frequency estimates from a

Table 4.III

SIMULATED MULTIPLE LINETRACKER DATA FOR SENSOR I FROM SCENARIO 1

SAMPLE NUMBER	TIME sec	TARGET NUMBER	FREQ. Hz.	BEARING (°)	SAMPLE NUMBER	TIME sec	TARGET NUMBER	FREQ. Hz.	BEARING (°)
1	5	1	149.65	360	24	115	2	149.85	14
2		3	150.15	52	25		1	149.45	26
3	15	2	149.95	356	26		3	150.05	34
4		1	149.55	0	27	125	2	149.85	15
5	25	2	149.95	359	28		1	149.45	26
6		1	149.55	10	29		3	150.05	51
7		3	150.15	43	30	135	2	149.85	9
8	35	1	149.55	12	31		1	149.45	28
9	45	2	149.95	14	32		3	150.05	39
10		1	149.55	16	33	145	2	149.85	0
11	55	2	149.95	6	34		1	149.45	30
12		1	149.55	18	35		3	150.05	41
13		3	150.15	45	36	155	2	149.85	14
14	65	2	149.95	3	37		1	149.45	30
15		1	149.55	19	38		3	150.05	62
16	75	2	149.95	4	39	165	2	149.85	24
17		1	149.45	23	40		1	149.45	31
18	85	2	149.85	5	41		3	150.05	40
19		1	149.45	24	42	175	2	149.75	22
20	95	1	149.45	24	43		1	149.45	30
21		3	150.05	48	44		3	150.05	42
22	105	2	149.85	4	45	185	1	149.45	35
23		1	149.45	28	46		3	150.05	33

linetracker's observation window, the presence of other signals close to the desired signal may cause the frequency estimates to skip in time from one signal to another. Such software restrictions on the frequency estimator could prevent the DIFAR sonobuoys from generating the type of theoretical, multiple target data sets presented in Table 4.III.

Due to these problems, it was decided to look at the power spectra with all of the signals present and no "OR-ing" to see if signals could be sorted from the noise in these spectra. The following technique has been used to simulate power spectra with multiple narrowband tones present.

To generate the simulated power spectra, the data generation program was first modified to furnish the simulated omnidirectional power spectra and the associated X and Y channel information instead of the simulated MAX-OR linetracker estimates. This allowed the spectra for one target's trajectory to be saved so it could later be merged with another target's set of spectra. Besides changing the output from the data generation program, the option was added to zero out all bins in the comb filter bank that contain only ambient noise powers before the individual target power spectra were output. With these two options, one set of simulated power spectra with both noise and signal present could be generated for one target. Next, power spectra that were zero filled except for the bins with true signal present could be generated for the remaining targets. These data sets could then be combined to produce simulated spectra that contained the narrowband tones of multiple targets and ambient noise.

The multiple target power spectra were generated in the following fashion which is illustrated in block form in

Figure 4.3. For one target in each of the two scenarios, a simulated set of power spectra was generated. The spectra for the frequency band simulated contained the signature of the target's narrowband tone as well as random, ambient noise. This was done for all three sensors' channels. Next, power spectra were generated for each of the remaining targets in each scenario that contained only the target's individual signature with all of the remaining frequency bins zero filled. Then, these spectra were merged to create the multiple narrowband signals and random ambient noise. For each frequency bin in the simulated frequency band, the simulated data in the omnidirectional channel, the X-channel and the Y-channel were merged for all of the targets involved in that scenario. After thresholding the omni spectra, frequency and bearing estimates are provided every 10 seconds for each bin that exceeds the threshold. These data are generated for each sensor for the duration of each scenario.

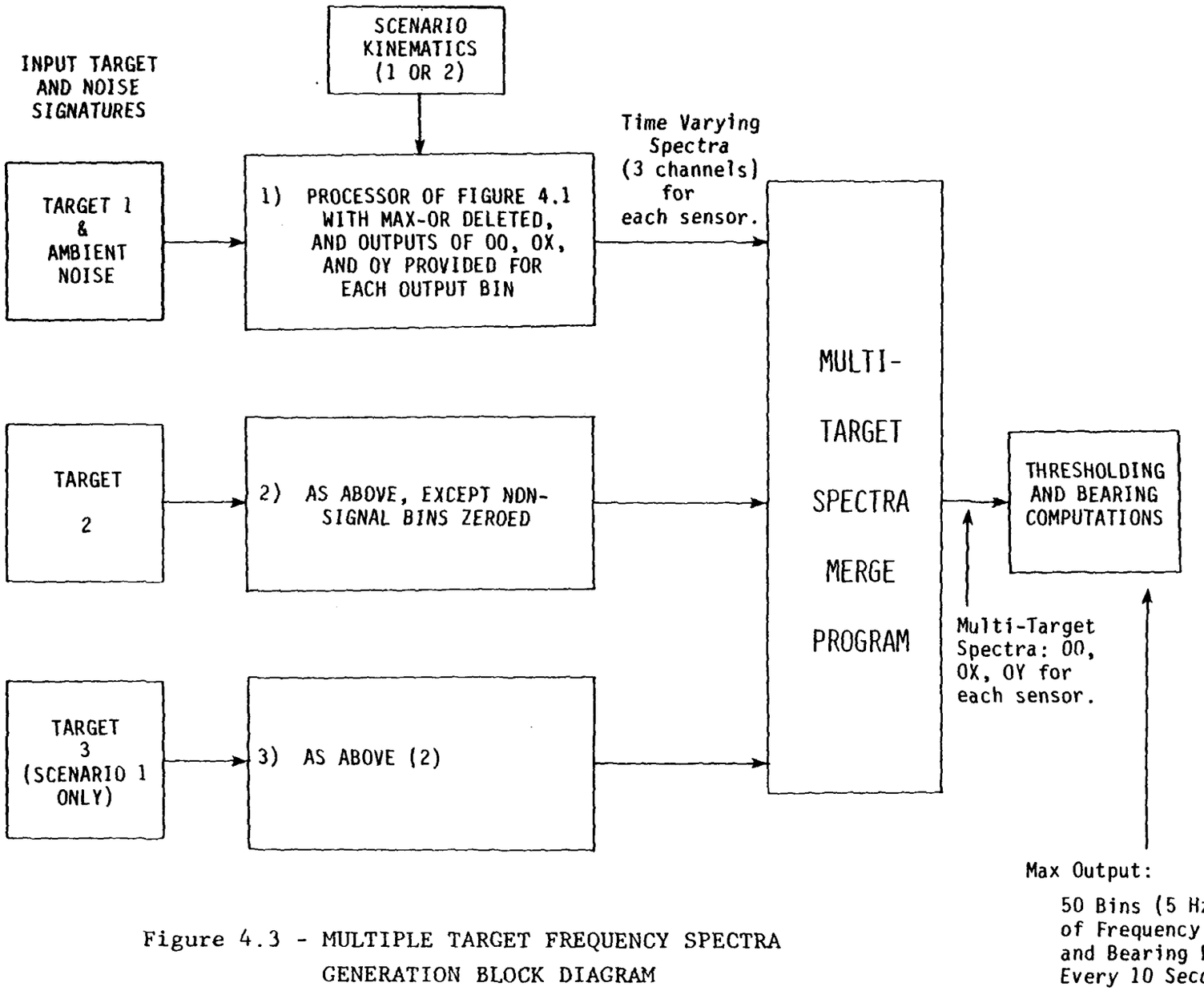


Figure 4.3 - MULTIPLE TARGET FREQUENCY SPECTRA GENERATION BLOCK DIAGRAM

5.0 CLUSTER ANALYSIS FOR MULTIPLE TARGET DATA SORTING

Cluster analysis is a field in numerical taxonomy which seeks to collect objects into natural groupings by objectively discriminating between arbitrary sets of attributes which describe these objects. Most of the developments in this field have been pioneered by researchers in the social, biological, and anthropological sciences. Three excellent sources of information about clustering techniques can be found in References 7, 8, and 9. Clustering techniques have proven to be useful in these fields for collecting items into natural groups that were often overlooked by researchers. It was suggested by Tracor that cluster analysis be investigated as a possible approach to the inherent problem of multi-target data sorting for the larger problem of multi-target sonobuoy target tracking. The concept suggested was that cluster analysis might be useful in identifying and separating intermixed measurements from multiple targets. Once input measurements were separated by target, it would then be possible to overcome multi-target initialization problems and it might also be possible to solve the multi-target localization problem with single target algorithms, each operating with measurements from only one target. Research on the application of cluster analysis to the data sorting problem for sonobuoy tracking is the subject of this section.

A major task for this contract called for a search to determine the optimum clustering procedure for separating data from multiple targets into individual data sets. For this study, one cluster program package consisting of several clustering techniques developed by the Department of Forestry and Outdoor Recreation at Utah State University was used (Reference 10). This program's techniques were designed

to group objects into hierarchical clusters. A second program was developed at Tracor from papers written by Ling (References 11 and 12) to cluster items into sets of natural groupings. These two programs formed the computational basis for this study.

The objective of the present effort was to explore the basic feasibility of performing acoustic data sorting with the techniques of cluster analysis. This required a multi-step process. First, as outlined above, the computational capabilities required for such research were acquired and developed. Then, the acoustic data sorting problem was analyzed to permit it to be approached by the methodologies of cluster analysis. This involved the definition of objects and attributes for the problem. Next, a preliminary study was conducted to narrow the scope of data normalization, cluster measures, and clustering algorithms that would be subjected to detailed study. With the range of variables suitably narrowed, the final part of the study was to evaluate, in some specific scenarios, the kind of performance that could be obtained from cluster analysis with respect to the data sorting problem.

The results obtained from this program of work are encouraging, but they are incomplete. Further research is indicated as being warranted. Specifically, the results show that cluster analysis can perform several acoustic data sorting functions, and that these functions should lend themselves to future automation. Positive results were obtained in connection with data outlier detection and removal, multi-target data sorting by target, and target data/noise sorting. It is felt that the results of this study establish that cluster analysis can be used successfully to perform all of these functions in the context of sonobuoy target tracking.

These results are provided by a methodology that simultaneously inspects all of the measured attributes of each data point and then groups data together, via fixed rules, which are most alike in terms of all the measured attributes. It could be stated that this computational formalism simply automates a process that is intuitively pleasing for acoustic data sorting; namely, group data that are similar in their physical measures such as frequency and bearing. Cluster analysis goes beyond intuition, however, in that it can handle an n-dimensional attribute vector as easily as it can a single sorting variable.

A serious drawback to the automated use of cluster anlaysis was indicated by the results of this study, however, and it appears to stem from the data normalization problem. As will be discussed in the remainder of this section, there is a cluster threshold that must be defined in order to obtain successful cluster separation of valid data and outliers, of multiple target data sets, or of valid target data and noise. How to set this threshold was not determined by the present work. This problem was clearly identified by the present research, but it remains unsolved. Any practical application of cluster analysis to acoustic data sorting must address this problem, but it was beyond the scope of this study, which has dealt with the more basic aspects of concept feasibility. In relation to Section 2 of this report, it should be noted that the data sorting studied here falls into the batch processing category. The concept should be expandable to sequential processing, however, by the future development of known techniques.

The remainder of this section is rather lengthy. Subsections 5.1, 5.2, 5.3 and 5.4 introduce information about various aspects of cluster analysis. Subsection 5.5 describes the preliminary work done to reduce the scope of the detailed

scenario evaluations. Subsection 5.6 discusses the application of cluster analysis to the single target data outlier removal problem. Input data for the algorithms are introduced here, as are the clustering tree diagrams. It is these diagrams that constitute the clustering algorithm's output at present. Which data samples are clustered, and at what confidence level they are grouped are indicated by these diagrams. Subsections 5.7 and 5.8 present very detailed scenario evaluation results. Subsection 5.7 addresses the application of clustering to multi-target data sorting with input data supplied by multiple linetrackers. Subsection 5.8 addresses the application of clustering to separating valid multi-target data from noise in frequency spectra data. In these two discussions, Subsections 5.7.1, 5.7.2, 5.8.1 and 5.8.2 contain considerable detail, and can be skipped over on first reading. Finally, Subsection 5.9 contains all the major conclusions reached about the feasibility of using cluster analysis for multi-target acoustic data sorting based on the results of this study.

5.1 Definition of Objects and Attributes for the Clustering Study

Cluster analysis requires that a group of objects be collected so that it may be determined which of these objects exhibits the most similarity between them. Associated with these objects is a set of attributes that is used to describe certain characteristics about the objects. The objects are to be clustered into natural groups based upon the descriptions provided by these attributes. For the current investigation, the objects consisted of a set of prospective acoustic signals that was to be separated from any ambient noise, and the remaining true data were to be clustered into data sets that should coincide with individual targets represented in the data. Initially, the attributes for each

prospective signal were chosen to consist of a data triplet represented by the time tag, bearing estimate and either the frequency estimate or its associated bin number in the comb filter bank. After a 360° to 0° discontinuity in bearing convention was encountered, it was decided to substitute the sine and cosine of the bearing estimate for the bearing estimate. This resulted in a set of attributes for each prospective signal that consisted of the time tag, sine of the bearing estimate, cosine of the bearing estimate and the frequency estimate. The possibility of using the SNR value at the receiver was considered as a fifth attribute, but the SNR values were found to fluctuate so wildly that they did not prove to be useful for data sorting.

5.2 Standardization of the Attributes

Before analyzing the results of the preliminary study, several other concepts regarding the clustering programs should be discussed. One point concerns standardizing the data in some fashion to produce better results in grouping the data. For the scenarios used, data were output at 10 second intervals. The frequency measurements for a given target varied by less than one Hz over the entire track and the bearing measurements varied by, at most, one radian over any track. The numerical difference in raw time units between successive measurements for an individual target is much larger than the numerical change in bearing units and frequency units. Because of this large difference, the clustering programs tended to group measurements by time tags rather than by individual targets when non-standardized data were used. CLUSTAR, the clustering package from Utah State University, contains five alternatives for standardizing data. The standardization techniques may be employed with individual attributes or may be used on all of the attributes at once.

The attributes may be standardized in the following manners (i refers to the individual attribute number, j to the data quadruplet index number):

(1) $X_{ij} - \bar{X}_i$

(2) X_{ij}/σ_i

(3) $(X_{ij} - \bar{X}_i)/\sigma_i$

(4) $X_{ij}/\max_j(X_{ij})$

(5) $\left(X_{ij} - \min_j(X_{ij})\right) / \left(\max_j(X_{ij}) - \min_j(X_{ij})\right)$

All of these methods were used in this study to determine the best standardization technique for our problem.

5.3 Resemblance Matrix

After data have been accumulated and either standardized or left alone, some measure of similarity or dissimilarity between the objects must be generated. In general, these measures are computed by either a similarity coefficient or a dissimilarity coefficient. When similarity coefficients are used, a large value for the coefficient for a pair of objects implies a high degree of similarity between the pair. Conversely, if dissimilarity coefficients are used, a large coefficient for a given pair implies a large degree of dissimilarity between the individuals. One of these similarity/dissimilarity coefficients is used to transform the

data matrix or the standardized data matrix into a resemblance matrix. CLUSTAR has seven different similarity/dissimilarity coefficients that may be used. These methods are named and described below. (NOTE: Subscripts j and k refer to object numbers, subscript i refers to a specific attribute.)

Method 1

correlation coefficient r_{jk}

$$r_{jk} = \frac{\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)(X_{ik} - \bar{X}_k)}{\left(\sum_{i=1}^{n} (X_{ij} - \bar{X}_j)^2 \sum_{i=1}^{n} (X_{ik} - \bar{X}_k)^2 \right)^{\frac{1}{2}}}$$

Method 2

average Euclidean distance d_{jk}

$$d_{jk} = \left(\sum_{i=1}^{n} (X_{ij} - X_{ik})^2 / n \right)^{\frac{1}{2}}$$

Method 3

vector dot product coefficient $\cos \theta_{jk}$

$$\cos \theta_{jk} = \frac{\sum_{i=1}^{n} X_{ij} X_{ik}}{\left(\sum_{i=1}^{n} X_{ij}^2 \right)^{\frac{1}{2}} \left(\sum_{i=1}^{n} X_{ik}^2 \right)^{\frac{1}{2}}}$$

Method 4

coefficient of shape difference z_{jk}

Let d_{jk} be the average of Euclidean distance

$$\text{Let } Q_{jk}^2 = \frac{1}{n^2}\left(\sum_{i=1}^{n} X_{ij} - \sum_{i=1}^{n} X_{ik}\right)^2$$

$$Z_{jk} = \frac{n}{n-1}\left(d_{jk}^2 - Q_{jk}^2\right)$$

Method 5

Clifford-Stephenson coefficient s_{jk}

$$s_{jk} = \frac{\sum_{i=1}^{n} |X_{ij} - X_{ik}|}{\sum_{i=1}^{n} (X_{ij} + X_{ik})}$$

Method 6

Canberra metric coefficient c_{jk}

$$c_{jk} = \frac{1}{n} \sum_{i=1}^{n} \frac{|X_{ij} - X_{ik}|}{(X_{ij} + X_{ik})}$$

Method 7

Bray-Curtis coefficient b_{jk}

$$b_{jk} = \frac{2\sum_{i=1}^{n} \min (X_{ij}, X_{ik})}{\sum_{i=1}^{n} (X_{ij} + X_{ik})}$$

Each of these seven measures have been tested to determine the optimal similarity/dissimilarity coefficient for our problems.

5.4 Clustering Algorithms

After a resemblance matrix has been computed for a given data set, some clustering technique must be used to determine how the data should be grouped. Several clustering techniques can be used, but one desires to use the technique which best clusters the data into groups that are appropriate for a given problem. For the simulated multi-target data used here, the correct grouping of data is known a priori; so, one knows what patterns he should be looking for from the clustering program's output. Knowing this information, tests can be run to determine the best clustering technique for separating data into individual target data for the multiple target tracking problem.

Five clustering techniques are currently available for separating the data. The four methods available with the CLUSTAR package include the single linkage method, the complete linkage method, the unweighted pair-group method using arithmetic averages (UPGMA), and Ward's method. Ling's papers describe a (k,r) clustering method. Each of these methods have individual characteristics which make them more desirable for specific problems. The single linkage method has also been called the nearest neighbor or the minimum method. A candidate member for an existing cluster has similarity to that cluster equal to its similarity to the nearest member within that cluster. This technique often produces straggly, chain-like clusters. Complete linkage, on the other hand, associates the similarity for a candidate point to an existing cluster to be equal to its similarity with the farthest member in the cluster. The complete linkage method is also known as the farthest neighbor method or the maximum method. Clusters produced by this method tend to be tight, hyperspherical,

discrete clusters. According to Sneath and Sokal (Reference 7), UPGMA is probably the most frequently used clustering strategy. UPGMA tries to group new points into an existing cluster by using an unweighted average similarity or dissimilarity within the cluster. Ward's method uses a within-group sum of squares objective function to decide in which cluster the point belongs. Ling describes his (k,r) clustering technique as a generalized single linkage algorithm which utilizes the k and r parameters to define the internal properties of a cluster. His (1,r), (i.e., k = 1), clustering algorithm, which corresponds to a classical, hierarchical, non-overlapping single linkage algorithm was developed for this study. All of these clustering methods have been evaluated in this investigation.

5.5 Optimal Clustering Techniques for the Multiple Target Problem

Bearing and frequency measurements, when viewed as functions of time for an individual target, appear as long chains for individual targets. These chains are rather smooth and continuous when plotted for non-maneuvering targets. Since, initially, only non-maneuvering trajectories are being used for this study, it seemed as though the single linkage methods would work best for this problem. Initial studies have tended to verify this preliminary judgement.

To determine the best combination of techniques for processing the data with the cluster analysis programs, simulated multiple linetracker data were generated which contained measurements for all three of the targets involved in scenario 1 (see Figure 4.1). The resulting multiple target data were processed with each of the possible combinations of processing options for data normalization, resemblance matrix

generation, and cluster generation. Since simulated data were used in this study, it was known a priori how the data should be properly sorted. With this knowledge and with the clustering results obtained from each of the processing combinations, one could determine the optimal configuration.

Initially, one of the four clustering programs from CLUSTAR was chosen to sort the data and non-standardized data were used to generate the resemblance matrix coefficients. Each of the seven similarity/dissimilarity coefficients were used to generate an individual resemblance matrix. Results from this data processing combination were evaluated and then another clustering method was used to re-evaluate the same resemblance matrix coefficients. The raw data were used in such a fashion until all possible combinations of resemblance matrix coefficients and clustering methods had been tested. As mentioned in Subsection 5.2, it was determined that the difference in units for the raw attributes, especially when time units were compared to frequency and bearing units, was much too drastic for any of the resemblance matrix-clustering methods to succeed. Therefore, it was decided to examine the possibilities of normalizing the attributes to improve the clustering results.

Next, alternative data normalizations were chosen to pre-process the data. For a given data normalization, resemblance matrices were generated for each of the seven similarity/dissimilarity coefficients and the results from processing these resemblance matrices with a given clustering algorithm were evaluated. After all the resemblance matrix options had been tested, a different clustering method was used to process each of the resemblance matrices. After these results were examined, another clustering method was picked and the process was repeated. This series of tests was continued

until all of the clustering methods had been evaluated. Then a new data normalization technique was used to standardize the data and the testing procedure was begun anew. This testing procedure continued until all possible combinations of data normalization, resemblance matrix generation, and clustering procedures had been evaluated.

The results from these exhaustive tests lead to our choice for the best clustering combination currently available for sorting multiple target data. CLUSTAR's single linkage clustering method outperformed all other methods when it used a resemblance matrix consisting of Euclidean distance dissimilarity coefficients for raw data that had been normalized by method 5 to force all of the attribute values to lie between 0 and 1. Several of the data normalization techniques such as methods 3 and 4 showed promise, but none performed as well as method 5. Similarly, some of the resemblance matrix options such as the vector dot product and coefficient of shape difference coefficients yielded reasonable results, but none of their results were found to be as good as results obtained with the Euclidean distance dissimilarity coefficients. As was previously stated, the single linkage clustering algorithm was expected to perform best of all the clustering algorithms due to the straggly, chain-like nature of the raw data. The complete linkage method tended to form initial small clusters well, but these clusters were not properly joined after these initial clusters were formed. None of the other clustering schemes worked as well as the single linkage algorithm.

After testing all of CLUSTAR's processing capabilities, Ling's (1,r) algorithm was tested. Ling's algorithm uses Euclidean distances for dissimilarity coefficients. He states (Reference 12) that the class of (1,r)

algorithms is identical to single linkage algorithms. In view of these two facts, it was decided to normalize all of the raw data to lie between 0 and 1 as above and compare the (1,r) results to those from CLUSTAR's single linkage algorithm. The clusters picked by the (1,r) algorithm were found to be identical to those determined by the single linkage algorithms. Because of the identical results in cluster formations, it was decided that only one of these two algorithms needed to be used in continuing our investigations. CLUSTAR's single linkage algorithm was chosen and its specific results are presented in the following sections.

5.6 Clustering to Remove Outliers from Single Target Data

Outlier removal is a problem that is most commonly associated with single target tracking problems. Outliers are defined as points in a measurement set that do not truly belong to the target being observed. Various factors can lead to outliers occurring in a data set. For instance a measuring device may actually detect a second target and mistakenly associate the measurement for this target with the measurements for the primary target. Other times, ambient noise may dominate the actual signal such that a measurement is generated for random noise rather than for an actual target signal. Sometimes hardware or software problems can lead to outliers being included in the data stream. Whatever causes these outliers to arise in the data, the problem is to recognize these points as outliers and then to eliminate them from the measurement set. If one attempts to track a target with data that contain several outliers, it becomes quite likely that the tracker either will not converge onto a valid solution or that it will eventually be thrown off track when it attempts to incorporate the outliers into its estimates.

Particularly for target tracking applications, it is very important that outliers be recognized and removed from the data so that accurate tracking solutions may be obtained for a particular target.

In the past, several approaches have been tried to alleviate the problems that arise when outliers occur in target data sets. One approach has been to smooth the data by prefiltering it before passing it on to a tracking algorithm. Another commonly used approach is to initialize the tracker as soon as possible with the initial data and then use the measurement prediction feature from least squares tracking algorithms to decide whether to accept or reject new data. With least squares tracking algorithms, future measurements can be predicted by the algorithm along with an associated variance for this predicted measurement. One commonly used approach for outlier removal is to reject any measurement that exceeds by more than three or four sigma the predicted measurement from the target tracking algorithm. Another possibility is to merely ignore the outlier removal problem and process all of the measurements as though all of them are valid observations. If only a few outliers are contained in the data, tracking estimates may not be too adversely affected by processing the outliers along with the true data. However, when significant numbers of outliers are processed by the target tracker, the tracking solutions will tend to diverge from the true track of the real target.

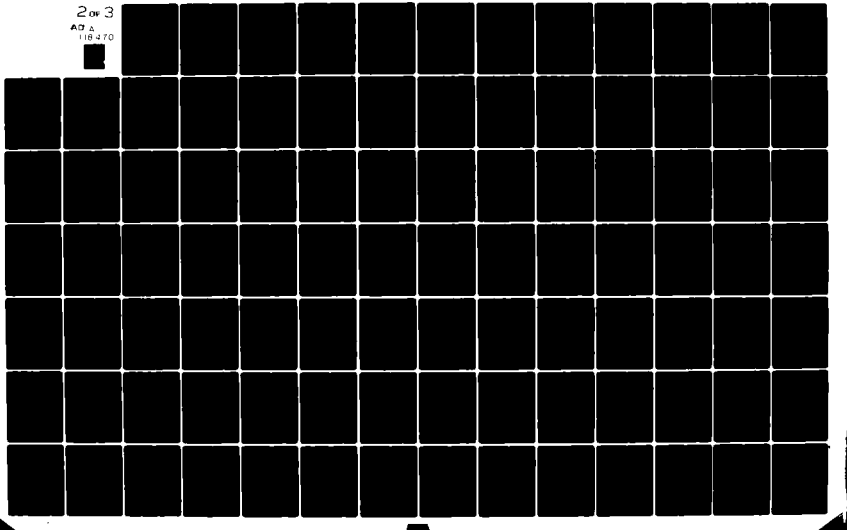
In this study, we have investigated the possibility of using single linkage clustering algorithms to initially identify and eventually eliminate outliers from true data for single target tracking applications. To test the possibilities for applying cluster analysis to the outlier removal problem, a simulated set of noisy data with random

noise peaks was generated in the following fashion. First, it was decided to generate simulated data for only target 1 in scenario 1 (see Figure 4.1). For this data, the mean SNR as determined at a distance one yard from the source was set to 76 dB. For the data that were simulated, the threshold level for minimum signal strength was set low enough to assure that frequency and bearing estimates were output for each measurement update, regardless of whether these estimates were true signal measurements or random noise measurements. Data were generated in this fashion for all three observing sensors in scenario 1. After generating the data in this fashion, efforts were then made to sort the data from each sensor in order to separate the true data from the random outliers.

The results from this preliminary investigation have been rather encouraging. For the two buoys where both true measurements and random noise measurements were present in the data, the tree diagrams output by the cluster analysis program indicated that the program could differentiate between the true signals and the random noise. However, for the third buoy, the signal was so strong that no random noise peaks were found in the measurements. For this case, the clustering algorithm separated the data into three separate clusters and then joined them together at high values for the dissimilarity coefficient. These three clusters proved to represent the three different frequency cells into which the measurements fell. As will be explained, it is felt that this unexpected problem was created by the normalization technique used to pre-process the data forcing small differences to be magnified many times greater than their true differences. Reviewing these results, it appears as though the data normalization question needs to be re-examined, but the overall results obtained demonstrate success of the concept and merit discussion.

AD-A118 470 TRACOR APPLIED SCIENCES AUSTIN TX F/G 15/4
HYBRID TRACKING ALGORITHM IMPROVEMENTS AND CLUSTER ANALYSIS MET--ETC(U)
FEB 82 D COOPER, G CORSER, T WILSON N00014-78-C-0670
UNCLASSIFIED TRACOR-T82-AU-9054-U NL

2 OF 3
AD A 118 470



Specific results for sorting the true signals from the outliers in the data gathered by sensors II and III will be presented here. The simulated data is presented in Table 5.I for sensor II. The corresponding tree diagram produced by the single linkage clustering algorithm is presented in Figure 5.1. The data and clustering tree diagram are not included for sensor III because the results were very similar to those for sensor II. In all of these clustering tree diagrams, the sample numbers of the candidate measurements (objects) are found on the vertical axis and the associated dissimilarity coefficients are found on the horizontal axis. In Figure 5.1, the true measurements are found in the upper portion of the tree diagram with tightly linked connections between these data. In the lower half of the tree diagram, loosely knit data are joined to the existing cluster at very high levels of dissimilarity which indicates that these remaining points have little resemblance to the points in the upper portion of the tree diagram. Reviewing the tree diagram in Figure 5.1, the true target measurements are found between samples 5 and 15. Beginning with sample 9, the remaining samples should be considered to be the outliers from this measurement set because their frequency and bearing estimates do not correctly correspond to a fairly smooth and continuous curve as should be expected for this non-moving trajectory. Similar behavior is found in the results for sensor III. From observing the tree diagrams, obvious cutoff points can be determined by big jumps in dissimilarity coefficients found after these points. The dissimilarity coefficients associated with these cutoffs are about 0.151 and 0.110 for sensors II and III, respectively. Furthermore, if one examines Table 5.I to separate the data as suggested by this interpretation of the tree diagram, one will indeed see that the outliers have been appropriately sorted from the true measurements. The SNR of

TABLE 5.I

SIMULATED DATA WITH OUTLIERS

FROM BUOY II FOR TARGET 1 OF SCENARIO 1

- - - - - - DATA MATRIX - - - - - -

INPUT FORMAT :
MATRIX NAME : DMRG
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS : 19
NUMBER OF ATTRIBUTES : 4
MISSING VALUE CODE : -9999.00
OUTPUT OPTION : 2

OBJECT #	DATA MATRIX ATTRIBUTE #			
	TIME 1	FREQ 2	COS θ 3	SIN θ 4
1	5.0000	148.0500	.9827	-.1853
2	15.0000	150.5500	-.3605	-.9328
3	25.0000	150.5500	-.8212	-.5707
4	35.0000	150.5500	-.6975	-.7166
5	45.0000	150.5500	-.4321	-.9018
6	55.0000	150.5500	-.3333	-.9428
7	65.0000	150.5500	-.2095	-.9778
8	75.0000	150.5500	.7187	.6953
9	85.0000	151.6500	-.9861	-.1660
10	95.0000	148.4500	.8279	-.5609
11	105.0000	151.8500	.8682	-.4963
12	115.0000	150.5500	-.4533	-.8914
13	125.0000	147.7500	-.3926	.9197
14	135.0000	149.7500	-.2720	.9623
15	145.0000	150.5500	-.4358	-.9001
16	155.0000	147.6500	.6849	.7286
17	165.0000	150.9500	.9438	-.3305
18	175.0000	151.0500	.7220	.6919
19	185.0000	150.1500	.9767	.2144

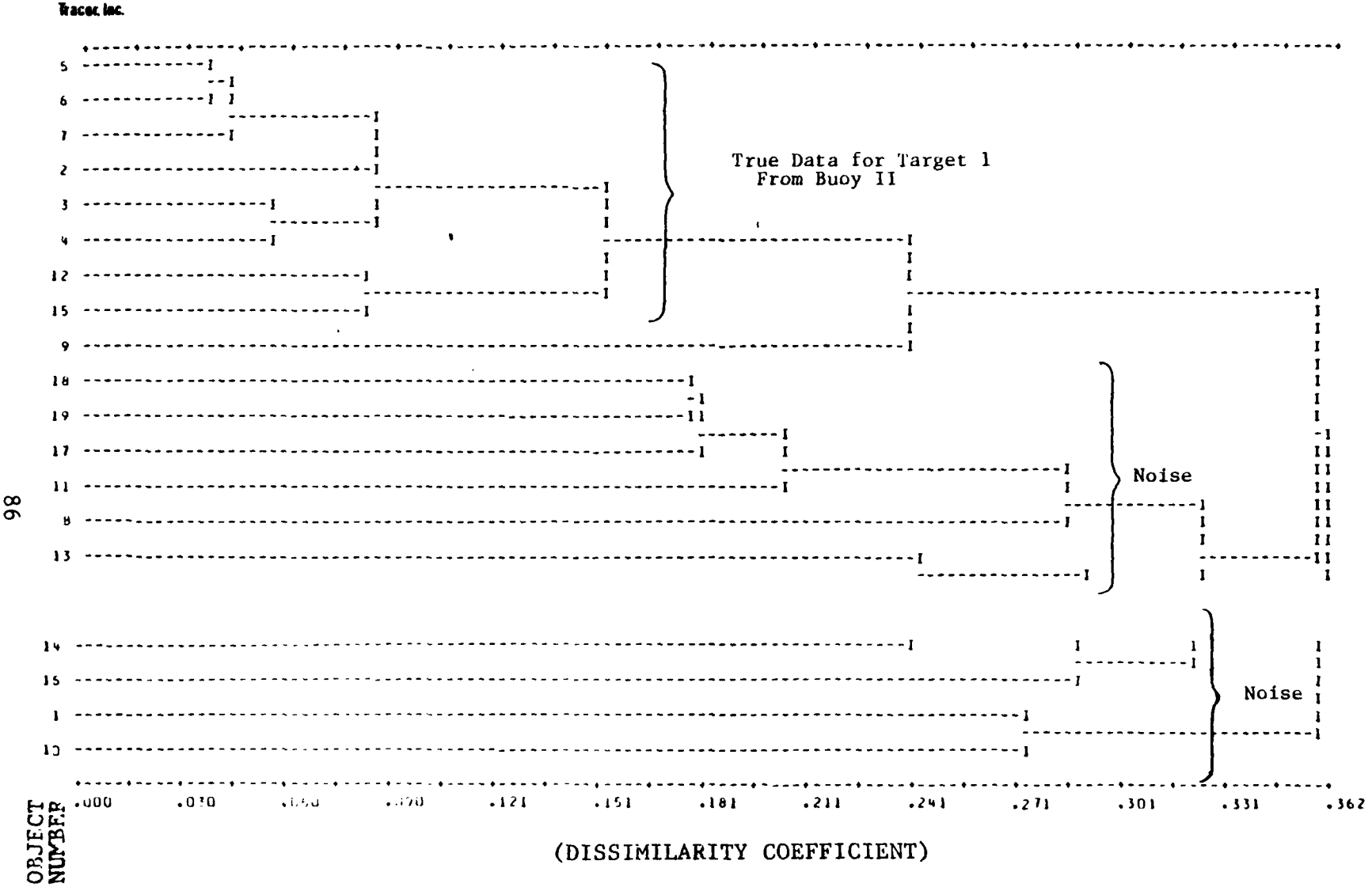


Figure 5.1 - CLUSTERING TREE DIAGRAM FOR DATA PLUS OUTLIERS FROM BUOY II FOR TARGET 1 OF SCENARIO 1

the measurements in these data sets varied from nearly -10 dB to over 4 dB. A five Hz frequency band was covered by a linetracker containing fifty cells. Conventionally, approximately a 0 dB threshold would be used to accept or reject the measurements. This approach would have rejected all of the noise measurements, but it also would have rejected some true measurements. However, when no data were rejected by the threshold test and the clustering algorithm was allowed to sort the data, the clustering algorithm correctly chose data whose detected SNR's were as low as -5 dB while it successfully rejected random noise signals as strong as -1.5 dB. This ability to intelligently compare data and choose true measurements while rejecting noise seems to be a vast improvement over thresholding data more to prevent false alarms than to select all of the possible true measurements actually produced by the signal processor.

A different response was found when cluster analysis was used to sort the data gathered by buoy I for target 1 of scenario 1. Target 1 traveled very close to buoy I throughout the length of this short scenario, so the propagation losses were never very large for this setting. Surprising initially, the data, listed in Table 5.II, were grouped into essentially three distinct clusters as can be seen in Figure 5.2. After reviewing the results, it was found that all three clusters coincided with the three distinct frequency measurements found in the data. The changing dynamics and geometries of this scenario forced the Doppler shifted frequency to appear in three different frequency bins during the observation period used for this scenario. Looking at Figure 5.2, samples 2 through 7 appear as one tightly knit cluster in the upper portion of the tree diagram. The frequency estimates for all of these samples were 149.55 Hz.

TABLE 5.II

SIMULATED DATA WITH OUTLIERS

FROM BUOY I FOR TARGET 1 OF SCENARIO 1

- - - - - - DATA MATRIX - - - - - -

INPUT FORMAT :
MATRIX NAME : DMRG
TYPE OF MATPIX : DATA
NUMBER OF OBJECTS : 19
NUMBER OF ATTRIBUTES : 4
MISSING VALUE CODE : -9999.00
OUTPUT OPTION : 2

- - - - - - DATA MATRIX - - - - - -

	1	2	3	4
1	5.0000	149.6500	.9999	-.0149
2	15.0000	149.5500	.9971	.0764
3	25.0000	149.5500	.9865	.1641
4	35.0000	149.5500	.9775	.2109
5	45.0000	149.5500	.9618	.2736
6	55.0000	149.5500	.9453	.3262
7	65.0000	149.5500	.9481	.3180
8	75.0000	149.4500	.9175	.3977
9	85.0000	149.4500	.9027	.4302
10	95.0000	149.4500	.9123	.4096
11	105.0000	149.4500	.8713	.4908
12	115.0000	149.4500	.8961	.4398
13	125.0000	149.4500	.9057	.4239
14	135.0000	149.4500	.8845	.4665
15	145.0000	149.4500	.8640	.5035
16	155.0000	149.4500	.8703	.4926
17	165.0000	149.4500	.8482	.5297
18	175.0000	149.4500	.8747	.4847
19	185.0000	149.4500	.7968	.6043

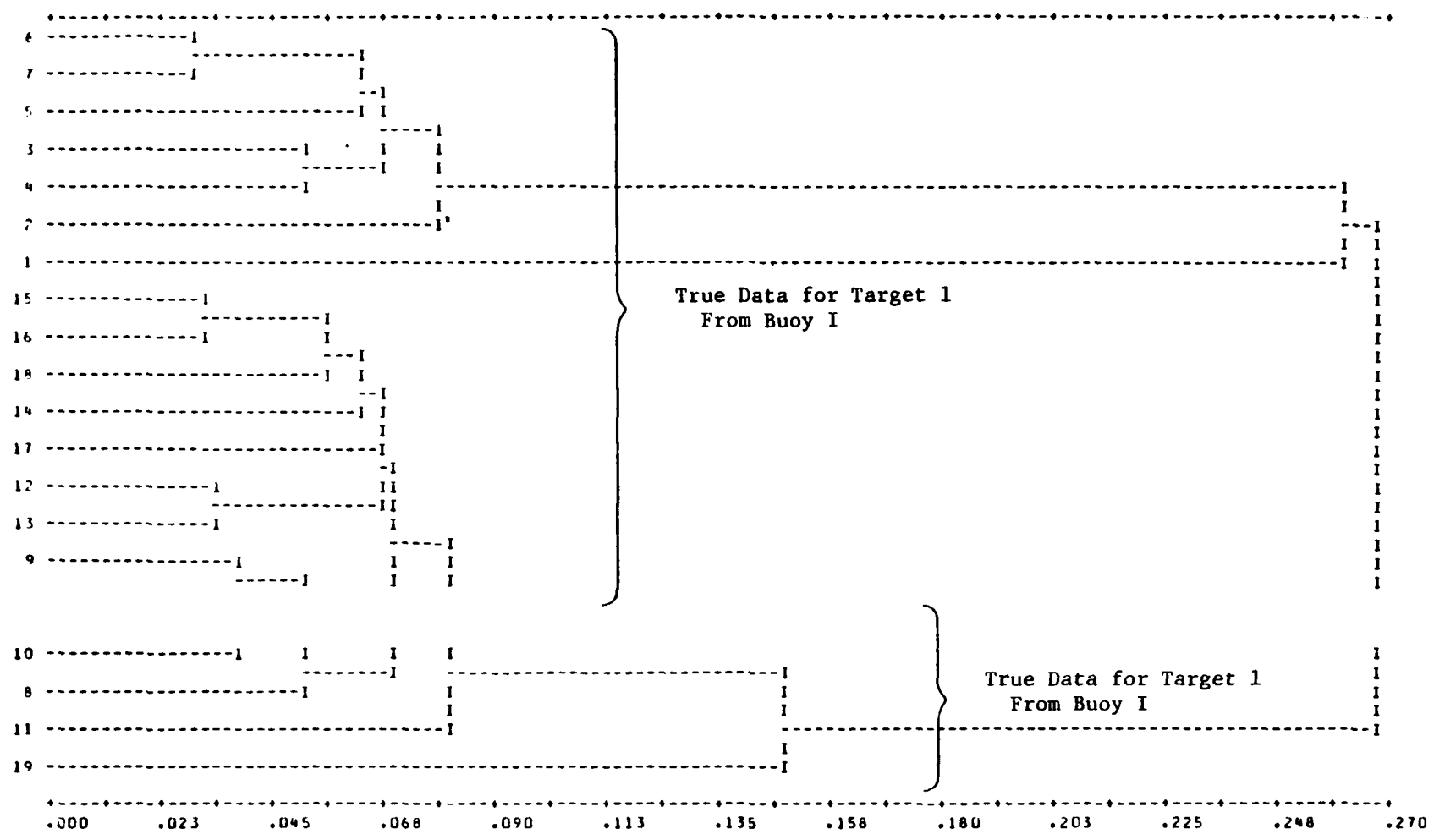


Figure 5.2 - CLUSTERING TREE DIAGRAM FOR DATA PLUS OUTLIERS FROM BUOY I FOR TARGET 1 OF SCENARIO 1

Following this cluster, a solitary cluster consisting of only sample 1 is found. The frequency estimate for sample 1 was 149.65 Hz. Following this cluster, the remaining samples 8 through 19 are found grouped into one large cluster. For each of these samples, the frequency estimates were 149.45 Hz. All three clusters are eventually merged into one cluster as should be the case for these data, but the clusters are joined at such high levels (approximately 0.270) relative to the other clusters, that one would probably assume they should not really be joined together. Unfortunately, an examination of the data shows that they are all true measurements and belong to the same target.

Intuitively, this data separation is disconcerting because we want the clustering algorithm to sort outliers but not to falsely sort the data from one source into multiple data sets. Careful examination of the raw data in Table 5.II would tend to indicate that the problem could lie with the data normalization technique used on the raw data. For the frequency and sin θ raw data from buoy I, the maximum difference between any two samples is only 0.2. Recalling the normalization method used to pre-process the data, this 0.2 difference appears in the denominator of the normalization equation. Instead of scaling any differences to be smaller, this denominator effectively magnifies any differences by a factor of five. On the other hand, if one reviews the raw data for buoy II in Table 5.I, the differences between the maximum and minimum values used by the normalization equations equal approximately 4.2 and 1.97 for frequency and sin θ, respectively. For buoy III, these differences will be approximately 3.9 and 0.7 for frequency and sin θ, respectively. Buoys II and III tend to normalize the raw data such that the numerical differences for the attributes are made smaller. Conversely, the differences for buoy I are magnified

after normalization because all of the raw data already have small variability. With common normalization scale factors for each attribute for all of the buoys, the results from Buoy I would be much better than that now seen in Figure 5.2. In fact, the true data cluster for Buoy I should be much tighter knit than the clusters of true data for the other two buoys. This hypothesis for explaining the discrepancies in the results for the three buoys has not yet been tested, but it makes sense intuitively. Clearly, the current data normalization technique seems to have some problems, but the single linkage clustering algorithm, nonetheless, shows promise for solving the outlier identification and removal problem.

5.7 Clustering to Sort Multiple Target, Multiple Linetracker Data

After examining the possibility of using cluster analysis to solve the outlier removal problem associated with single target data, the use of cluster analysis to sort data for the multiple target tracking problem was investigated. For the multiple target problems where only passsive DIFAR data will be used, no a priori knowledge of how many targets are present or what measurement values to expect will be available. An approach such as cluster analysis which looks for natural trends or natural groups without assumptions could be a reasonable approach to this problem. After noting the success with sorting simulated weak signals from random noise, a natural progression would be to use cluster analysis for data sorting in the multiple target tracking problem.

In this subsection, only simulated multiple linetracker data as described in Subsection 4.2 were used for the data sorting study. The 4-tuples consisting of the time

tag, frequency estimate, and the cosine as well as the sine of the bearing estimate were used to describe each detected signal. Idealized linetracker data were simulated which ignored the possibility of signal interference from other sources when the measurement estimates were generated. In this section, the merged, multiple target linetracker data for each sonobuoy are examined to determine whether cluster analysis can be used to separate this data into distinct sets of individual target data. The raw data and clustering tree diagram for sonobuoy I of scenario 1 are included in this subsection, but the raw data and tree diagrams for sonobuoys II and III of scenario 1 as well as those for all three sonobuoys of scenario 2 have been excluded from this report to streamline the following discussions.

5.7.1 Multiple Linetracker Cluster Results for Scenario 1 - The results for each of the three sonobuoys from scenario 1 are discussed here. All of the data were generated in the normal fashion with a threshold of 0 dB used to determine whether to accept or reject a measurement estimate. Each target in the simulation transmitted only one narrowband tone at 150 Hz. Originally, a mean SNR value of 76 dB as measured one yard from the target was used to generate the measurements. However, difficulties were encountered in gathering sufficient data from all the targets for the clustering algorithm to effectively sort the data, so the transmitted SNR level was increased to 80 dB for all three targets. Simulated linetracker data for the targets were generated and then merged. In all further discussions, a sample will be denoted as an outlier when its frequency estimate varies too drastically to fit in the rather continuous, chain-like curve expected for non-maneuvering

targets. When necessary, drastic changes in bearing characteristics have also been considered in labeling a sample to be an outlier.

The simulated multiple target data for buoy I is presented in Table 5.III. A three-dimensional representation of these raw data is provided in Figure 5.3. In this plot, the curves represent the true, uncorrupted measurements that correspond to the actual dynamics and geometries of scenario 1. The pluses found close to these curves represent the simulated, noisy, non-Gaussian measurements produced for this simulation. The corresponding tree diagram output by the cluster analysis program is shown in Figure 5.4.

Looking at the tree diagram in Figure 5.4 for buoy I, one can see that there exists many different levels at which single data points and small clusters are joined. Eventually, all the clusters are linked into a single cluster. However, if one goes down a few levels from the level where all of the data are linked, the data sets for each individual target may be found in three separate clusters. After looking at the clusters and knowing from our data simulation which data points belong together, it is evident that the data for target 2 are contained in the upper portion of the tree diagram between samples 34 and 1. In the middle portion of this tree between samples 24 and 5, the data for target 1 are found. Data for target 3 are found in the lower portion of the tree diagram between samples 32 and 2. The very last sample, number 38, nominally belongs to target 3. However, the clustering tree diagram indicates great difficulty was encountered in linking this sample with any of the other data. A re-examination of the data shows that sample number 38 contains a frequency estimate that agrees with the adjoining frequency

TABLE 5.III

MULTIPLE LINETRACKER DATA FROM BUOY I OF SCENARIO 1

- - - - - - DATA MATRIX - - - - - -

INPUT FORMAT :
MATRIX NAME : PAR1
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS : 46
NUMBER OF ATTRIBUTES : 4
MISSING VALUE CODE : -9999.00
OUTPUT OPTION : 2

- - - - - - DATA MATRIX - - - - - -

	1	2	3	4
1	5.0000	149.6500	1.0000	-.0041
2	5.0000	150.1500	.6152	.7884
3	15.0000	149.9500	.9980	-.0633
4	15.0000	149.5500	.9961	.0881
5	25.0000	149.9500	.9999	-.0167
6	25.0000	149.5500	.9861	.1661
7	25.0000	150.1500	.7314	.6820
8	35.0000	149.5500	.9767	.2145
9	45.0000	149.9500	.9713	.2361
10	45.0000	149.5500	.9629	.2701
11	55.0000	149.9500	.9942	.1077
12	55.0000	149.5500	.9464	.3170
13	55.0000	150.1500	.7061	.7081
14	65.0000	149.9500	.9989	.0471
15	65.0000	149.5500	.9462	.3235
16	75.0000	149.9500	.9972	.0753
17	75.0000	149.4500	.9239	.3849
18	85.0000	149.6500	.9966	.0824
19	85.0000	149.4500	.9107	.4142
20	95.0000	149.4500	.9107	.4065
21	95.0000	150.0500	.6695	.7428
22	105.0000	149.6500	.9977	.0662
23	105.0000	149.4500	.9442	.4677
24	115.0000	149.6500	.9711	.2389
25	115.0000	149.4500	.8976	.4407
26	115.0000	150.0500	.8297	.5582
27	125.0000	149.6500	.9665	.2566
28	125.0000	149.4500	.8992	.4375
29	125.0000	150.0500	.6346	.7728

TABLE 5.III --- Concluded

30	135.0000	149.8500	.9863	.1650
31	135.0000	149.4500	.8938	.4678
32	135.0000	150.0500	.7747	.6323
33	145.0000	149.8500	1.0000	.0000
34	145.0000	149.4500	.8692	.4945
35	145.0000	150.0500	.7507	.6607
36	155.0000	149.8500	.9894	.2455
37	155.0000	149.4500	.8701	.4924
38	155.0000	150.0500	.7515	.6587
39	165.0000	149.8500	.9134	.4071
40	165.0000	149.4500	.8570	.5152
41	165.0000	150.0500	.7714	.6363
42	175.0000	149.7500	.9259	.3777
43	175.0000	149.4500	.8692	.4944
44	175.0000	150.0500	.7867	.6740
45	185.0000	149.4500	.8197	.5727
46	185.0000	150.0500	.8426	.5385

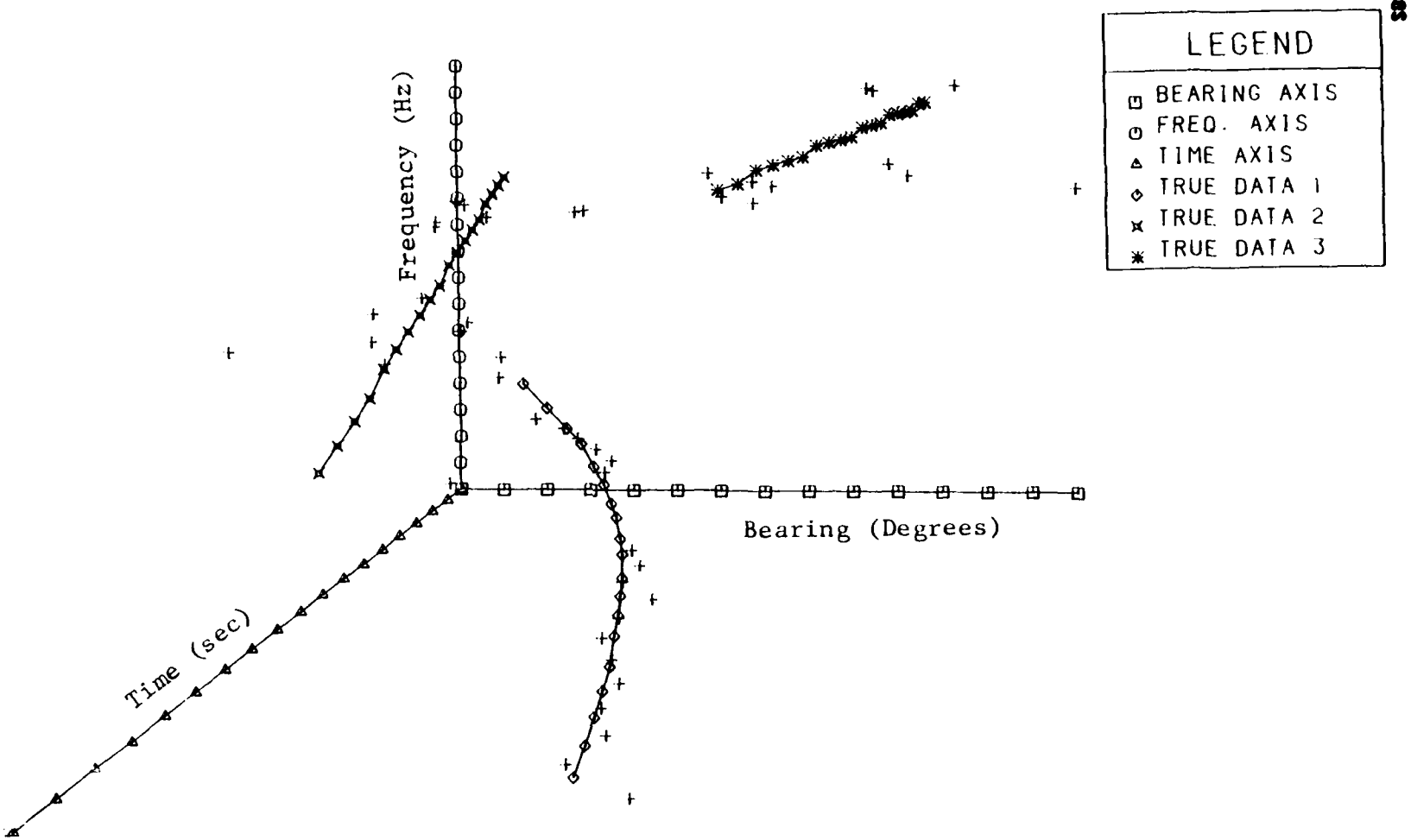


Figure 5.3 - True and Simulated Frequency vs. Bearing vs. Time Plot for all Three Targets Observed by Sensor I of Scenario 1

TRACOR

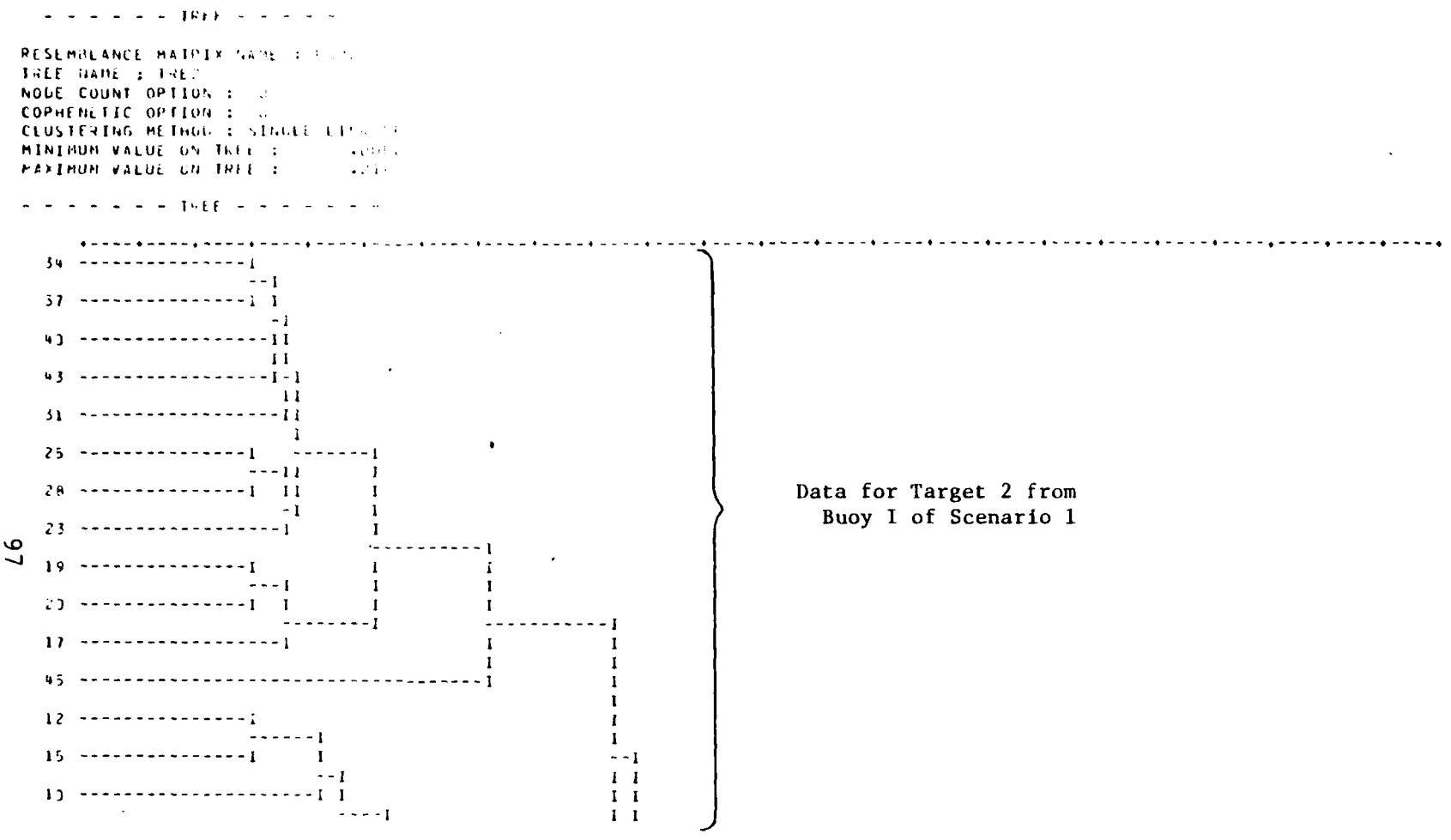


Figure 5.4 - CLUSTERING TREE DIAGRAM FOR MULTIPLE LINETRACKER DATA FROM BUOY I OF SCENARIO 1

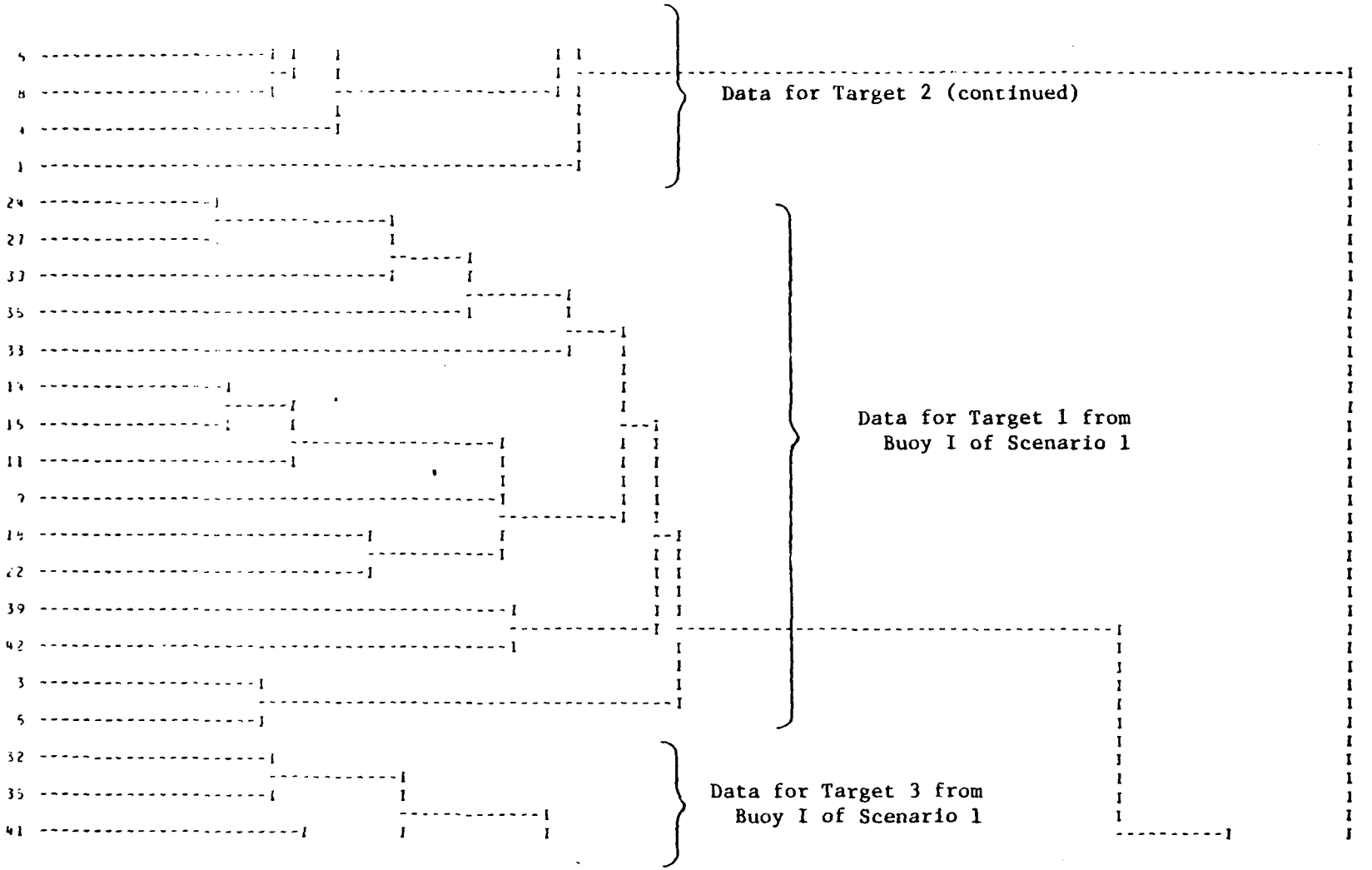


Figure 5.4 -- Continued

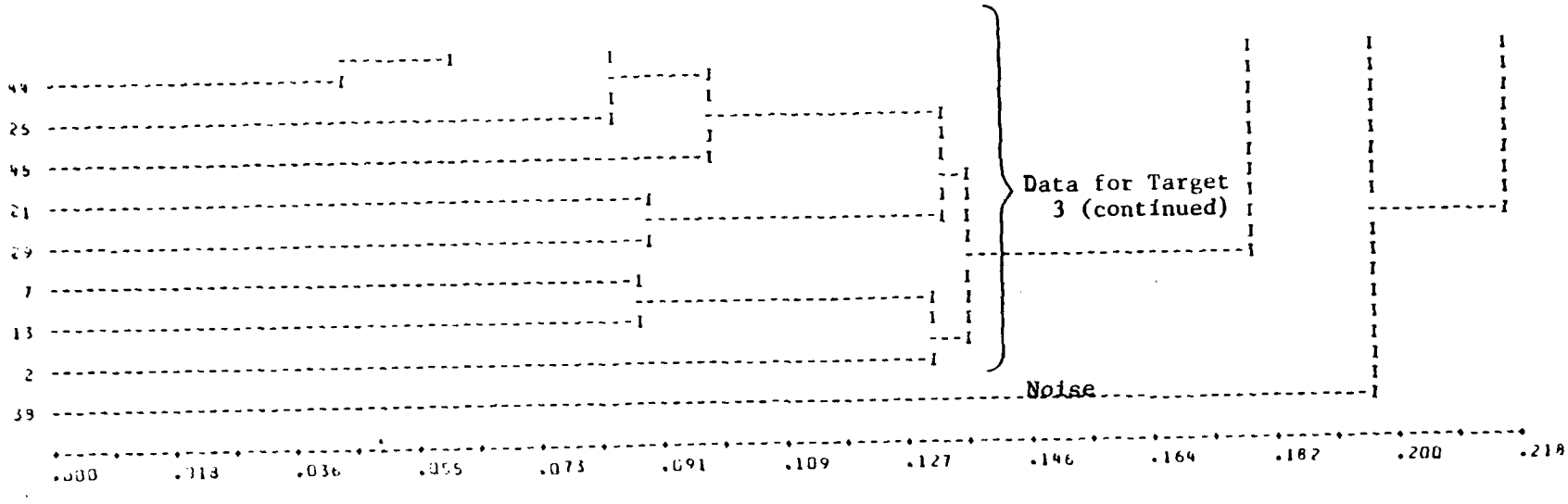


Figure 5.4 -- Concluded

estimates for target 3, but a close look at the cosine and sine of the bearing estimate for sample 38 shows that this bearing estimate differs substantially from the adjoining bearing estimates for target 3. Looking at the clustering tree diagram and the actual measurement values, it appears as though sample 38 should best be labeled as an outlier and removed from the data set. When sample 38 is eliminated as an outlier, the tree diagram can then be interpreted as correctly sorting the data into individual target data sets when a threshold level of approximately 0.140 is used to halt the linking of the clusters. If clusters are linked for dissimilarity coefficient values smaller than 0.140, the data will be correctly sorted into three different data sets, each of which corresponds to one of the three targets used in this simulation.

The tree diagram clustering results for the merged linetracker data for buoy II of scenario 1 exhibited the same behavior found in Figure 5.4 As always, the tree diagram showed all data to eventually be linked into one conglomerate cluster. However, when the tree diagram was reduced to the point where only three smaller clusters plus one data point were found, the data for each individual target were found to be correctly sorted. In the upper portion of the tree diagram, data for target 3 were found. Following this clump, the next six samples formed a cluster which contained the data for target 2. Except for the last sample in this tree diagram, the remaining data were grouped into a cluster of data which corresponded to target 1. Again, the last sample appeared to be an outlier, so it was eliminated rather than included with any of the other data. The frequency estimate for this last sample corresponded to the data for target 2, but the bearing estimate was so poor in comparison to the rest of the data for target 2 that it could not be included into this data set. If

a threshold of approximately 0.182 for the resemblance matrix coefficient was used to decide when to stop joining clusters together, the multiple target linetracker data for buoy II would be correctly grouped into three sets of data. Each of the three clusters corresponds to an individual target. The last sample in the tree diagram is an outlier which should be eliminated from this data stream.

Lastly, the results from scenario 1 for buoy III will be analyzed. Reviewing the tree diagram for this sonobuoy, no obvious outliers were found in the data. If one looked for the clustering level where all the data are grouped into three separate clusters, the correctly sorted individual data sets for each of the three targets in this scenario are found. Looking at the results from this tree diagram, it could be seen that the threshold value appropriate for this case would be approximately 0.130 for the dissimilarity coefficient. In conclusion, if these clustering results are appropriately analyzed and interpreted, the cluster analysis approach has been shown to provide a viable means for sorting multiple linetracker data into single target data sets for the multiple target problem.

5.7.2 Multiple Linetracker Cluster Results for Scenario 2 - The results from using cluster analysis to sort multiple linetracker data for scenario 2 are discussed in this subsection. Recall from Figure 4.2 that this scenario consisted of two targets that traveled parallel paths with identical velocities. This trajectory was expected to create ambiguities particularly for buoys I and III due to the strong similarities in both Doppler shifted frequency and bearing estimates that would be detected by these sensors. From this pathological case, some bounds could be established on the

sensitivity of the data sorting by cluster analysis to strong similarities in signal characteristics from two different sources. In this section, only the results from successful data sorting runs will be presented. As will be detailed in the succeeding discussion, signals from two different sources that are any more similar than the bounds established here will most likely be inseparable by the cluster analysis approach to data sorting.

The case where successful data sorting was first accomplished for buoy I occurred when the unshifted center frequencies transmitted by the two targets were separated by 0.5 Hz. Attempts were made to sort data when the center frequencies were separated by 0.0, 0.1, 0.2, 0.3 and 0.4 Hz, but the single linkage clustering algorithm could not suitably sort the data for these five cases because there was too little difference in the attributes between the two signals. Since the bearings could not be changed for this scenario, it was decided to vary the transmitted center frequency for the two targets until the data could be suitably sorted. The upper portion of the tree diagram for this case contained the sorted data for target 1. All but the last three samples of the remaining half of the tree diagram contained the data for target 2. The last three samples in the tree diagram were again considered to be outliers and were eliminated from the data stream. The dissimilarity coefficient value associated with this cutoff level was approximately 0.130.

Not so surprisingly, the clustering results for the multiple linetracker data from buoy II of scenario 2 were different from the results presented above. For buoy II, the two targets are moving toward the sonobuoy rather than past it as is the case for buoys I and III. With the targets moving at

this sensor at the same speed and same heading, the Doppler shifts will be the same but the separation in the bearing measurements will be larger than for buoys II and III. For buoy II, a difference of only 0.1 Hz between the transmitted center frequency values for the two targets was sufficient for the clustering algorithm to sort the data. When there was no difference in the transmitted center frequency values for the two targets, the clustering algorithm failed to adequately sort the data. The upper half of the tree diagram for buoy II contained all of the data for target 1. The remaining lower half of the tree diagram contained the data for target 2. For this tree diagram, there are a few samples that were more dissimilar than the other samples in the two clusters, but no obvious outliers could be found. For this tree diagram, a threshold cutoff of approximately 0.292 would result in two well defined clusters which contained data for the two targets found in scenario 2.

The results from clusterng the data for sonobuoy III were expected to be fairly similar to the results for sonobuoy I, and this proved to be the case. The transmitted center frequencies for the two targets had to be separated by at least 0.4 Hz for the clustering technique to properly sort the data. This is 0.1 Hz closer than the results from buoy I, but this difference is not considered to be significant. Once again, when the data are reduced to two clusters rather than one, each of the resulting clusters contains data for an individual target. A cutoff point of 0.133 for the dissimilarity coefficient would result in the correct sorting of the data into two sets of individual target data for buoy III.

5.7.3 Conclusions from Using the Single Linkage Clustering Algorithm to Sort Simulated Multiple Linetracker Data - The results from applying cluster analysis techniques to data sorting for multiple linetracker data have been quite encouraging. For scenario 1, three identifiable clusters which contained the data for the three targets could easily be found if the observer knew in advance to search for only three clusters. The results from the second scenario indicate that there are limitations as to how similar the data can be before the clustering algorithm can successfully sort the data into individual target data sets. Either the frequency or the bearing measurements or both of these measurements must have identifiable differences that are not lost in random noise before cluster analysis can succeed in separating the data. Unfortunately, no hard fast rule for determining a threshold level can be established from these results to decide when the joining of clusters should be stopped by the single linkage clustering algorithm. Simply for the data sorting problem, this threshold level varies from as low as 0.129 to as high as 0.292. In the outlier removal study, this threshold level varied from 0.101 to 0.270. Obviously, this threshold value is a dynamic parameter that depends strongly on the data normalization technique and that now varies from one application to the next. Without any means to determine or fix this threshold value a priori, it is impossible to automate this cluster analysis procedure so it could be used without any human decisions being required. The information is available in the clustering tree diagrams as has been shown in the previous discussions, but the question of automating and properly interpreting the results from this process when no a priori information is available for the data still remains a very troubling problem.

5.8 Clustering Frequency Spectra Data: Establishing Frequency Tracks

After seeing the qualified success obtained from using cluster analysis to remove outliers and sort multiple target linetracker data, it was decided to examine the possibility of using cluster analysis for one more application. The single linkage algorithm might be used to analyze simulated DIFAR frequency spectra to separate the signals from the noise found in the spectra. If so, the clustering approach could recognize either single tones or multiple tones found in the frequency spectra instead of recognizing only the single strongest tone as the MAX-OR processor does. If multiple signals could be recognized with this approach, then the need for multiple linetrackers to track multiple frequency lines could be eliminated. Furthermore, some of the restrictions might be relaxed on how close these multiple tones could be in the frequency spectra before they could be separated. The preliminary results have been encouraging and have shown that this approach can sort the signal data from most of the random ambient noise. However, the data could not also be sorted into individual target sets with this approach.

5.8.1 Results from Clustering Multiple Target Frequency Spectra for Scenario 1 - The table of the simulated multiple target frequency spectra data for buoy I of scenario 1 is presented in Table 5.IV. The first column in this table is the sample number assigned to that prospective measurement. The next column, labeled "1" in the table, is the time tag of the measurement. Following the time tag is the frequency bin number for the simulated measurement. The last two columns

TABLE 5.IV

SIMULATED MULTIPLE TARGET FREQUENCY SPECTRA

FROM BUOY I OF SCENARIO 1

- - - - - - DATA MATRIX - - - - - -

INPUT FORMAT :
MATRIX NAME : PWR1
TYPE OF MATRIX : DATA
NUMBER OF OBJECTS : 120
NUMBER OF ATTRIBUTES : 4
MISSING VALUE CODE : -9999.00
OUTPUT OPTION : 2

- - - - - - DATA MATRIX - - - - - -

	1	2	3	4
1	5.0000	21.0000	.9989	-.0476
2	5.0000	22.0000	1.0000	-.0241
3	5.0000	25.0000	.9966	-.0820
4	5.0000	26.0000	.8952	-.4457
5	5.0000	28.0000	.6363	.7715
6	5.0000	29.0000	-.6213	-.7836
7	15.0000	13.0000	-.2291	.9734
8	15.0000	21.0000	.9961	.0881
9	15.0000	25.0000	.9983	-.0579
10	15.0000	28.0000	.7376	.6752
11	25.0000	18.0000	-.4972	.8733
12	25.0000	21.0000	.9961	.1461
13	25.0000	24.0000	.3243	-.9460
14	25.0000	25.0000	.9972	-.0780
15	25.0000	28.0000	.5682	.8229
16	35.0000	21.0000	.9767	.2145
17	35.0000	25.0000	.9999	.0116
18	35.0000	28.0000	.5822	.8131
19	35.0000	42.0000	-.9986	-.0524
20	35.0000	43.0000	.2087	-.9735
21	45.0000	10.0000	-.7119	-.7022
22	45.0000	21.0000	.9623	.2701
23	45.0000	24.0000	-.4145	.9100
24	45.0000	25.0000	.9965	.0839
25	45.0000	28.0000	.7658	.6431
26	45.0000	50.0000	-.9329	-.3601
27	55.0000	17.0000	-.1616	.9869
28	55.0000	21.0000	.9484	.3170
29	55.0000	25.0000	.8835	-.4684

TABLE 5.IV --- Continued

30	55.0000	26.0000	-.8181	-.5750
31	55.0000	28.0000	.5511	.8344
32	55.0000	29.0000	-.8911	-.4541
33	55.0000	31.0000	-.6807	.7326
34	55.0000	39.0000	.4241	.9056
35	55.0000	45.0000	.7176	.6964
36	65.0000	13.0000	.3475	-.9377
37	65.0000	14.0000	-.6014	.7989
38	65.0000	16.0000	.6046	.7965
39	65.0000	21.0000	.9462	.3235
40	65.0000	23.0000	.9854	.1701
41	65.0000	25.0000	.9829	.1339
42	65.0000	28.0000	.7543	.6566
43	65.0000	37.0000	.9538	-.3006
44	65.0000	47.0000	.9434	.3317
45	75.0000	9.0000	.8734	-.4871
46	75.0000	14.0000	.7413	-.6711
47	75.0000	20.0000	.9230	.3849
48	75.0000	21.0000	.9306	.3661
49	75.0000	25.0000	.9931	.1168
50	75.0000	27.0000	.7584	.6518
51	75.0000	29.0000	.7046	.7096
52	75.0000	44.0000	.4007	-.9162
53	85.0000	8.0000	-.7009	-.7132
54	85.0000	12.0000	.9998	.0210
55	85.0000	18.0000	.6484	.7613
56	85.0000	20.0000	.9103	.4140
57	85.0000	24.0000	1.0000	-.0071
58	85.0000	25.0000	.9621	.2728
59	85.0000	27.0000	.7897	.6134
60	85.0000	28.0000	-.5912	-.8066
61	85.0000	31.0000	.5755	-.8178
62	85.0000	47.0000	.9874	-.1580
63	95.0000	16.0000	.9807	-.1955
64	95.0000	20.0000	.9127	.4085
65	95.0000	23.0000	.2741	-.9617
66	95.0000	25.0000	.3415	-.9399
67	95.0000	27.0000	.6977	.7164
68	105.0000	12.0000	-.5487	.8360
69	105.0000	20.0000	.8842	.4670
70	105.0000	27.0000	.4389	.8985
71	105.0000	34.0000	.8546	-.5193
72	105.0000	45.0000	-.9735	.2285

TABLE 5.IV -- Concluded

73	115.0000	5.0000	.4096	-.9123
74	115.0000	20.0000	.8976	.4407
75	115.0000	24.0000	.9987	-.0511
76	115.0000	25.0000	.8943	.4474
77	115.0000	27.0000	.7384	.6744
78	125.0000	14.0000	.6036	-.7973
79	125.0000	15.0000	.9943	.1062
80	125.0000	20.0000	.8992	.4375
81	125.0000	24.0000	.9934	.1148
82	125.0000	27.0000	.5919	.8060
83	135.0000	14.0000	-.7164	-.6977
84	135.0000	16.0000	-1.0000	.0004
85	135.0000	20.0000	.8933	.4678
86	135.0000	24.0000	.9659	.2589
87	135.0000	27.0000	.7576	.6528
88	135.0000	48.0000	-.2223	.9750
89	145.0000	6.0000	.0232	-.9997
90	145.0000	20.0000	.8692	.4945
91	145.0000	24.0000	.9940	.1092
92	145.0000	26.0000	.5615	-.8275
93	145.0000	27.0000	.6360	.7717
94	145.0000	30.0000	.3086	.9512
95	145.0000	42.0000	.7106	.7036
96	155.0000	1.0000	-.7663	-.6425
97	155.0000	20.0000	.8701	.4928
98	155.0000	24.0000	.9162	.4007
99	155.0000	27.0000	.7185	.6956
100	165.0000	20.0000	.8572	.5152
101	165.0000	24.0000	.9841	.1776
102	165.0000	25.0000	.5384	.8427
103	165.0000	27.0000	.7629	.6466
104	165.0000	33.0000	-.8322	.5545
105	165.0000	39.0000	-.6962	-.7274
106	165.0000	49.0000	-.4787	-.8782
107	175.0000	7.0000	-.4027	-.9153
108	175.0000	18.0000	-.1935	-.9811
109	175.0000	20.0000	.8692	.4944
110	175.0000	23.0000	.9805	.1965
111	175.0000	24.0000	.9915	.1313
112	175.0000	27.0000	.7402	.6700
113	175.0000	42.0000	.9848	-.1739
114	185.0000	3.0000	.1062	.9943
115	185.0000	10.0000	-.9871	.1599
116	185.0000	15.0000	.4275	-.9040
117	185.0000	20.0000	.8197	.5727
118	185.0000	23.0000	.9704	.2416
119	185.0000	27.0000	.7052	.7090
120	185.0000	30.0000	-.0542	-.9985

contain the cosine and sine, respectively, of the bearing estimate associated with the prospective measurement. The data for this buoy were generated in the manner described in Subsection 4.3. The associated three dimensional plot of the omnidirectional power versus frequency as a function of time for the three targets observed by sensor I of scenario 1 is shown in Figure 5.5. For this simulation, a five Hz frequency band was covered by 50 cells from a comb filter bank. This observed frequency band and the quadruplet of attributes assigned to each prospective measurement will remain the same for all the simulated multiple target frequency spectra generated for both scenarios 1 and 2. Note that multiple frequency estimates are generated at each time, and that they can lie anywhere within this five Hz band. As for Subsection 5.7, outliers were determined primarily by unacceptable discontinuities in the frequency estimates as functions of time. Where it proved to be useful, drastic variations in bearing estimates were also used to label data samples as outliers.

The tree diagram of the single linkage clustering algorithm output for the data in Table 5.IV is presented in Figure 5.6. The true signal data for this scenario are all found in the upper portion of this tree diagram. All of the real signal data are found between samples 110 and 70 in the tree diagram. From samples 79 on down, only random ambient noise is found. Between samples 110 and 70, two obvious outliers are found in samples 40 and 102. Other possible outliers may exist in this data, but these two samples are the most obvious ones because their frequency bin numbers do not correctly correspond to any of the true data for this time frame. Closer examination of Figure 5.6 shows clusters of partial data sets for each target. The data between samples

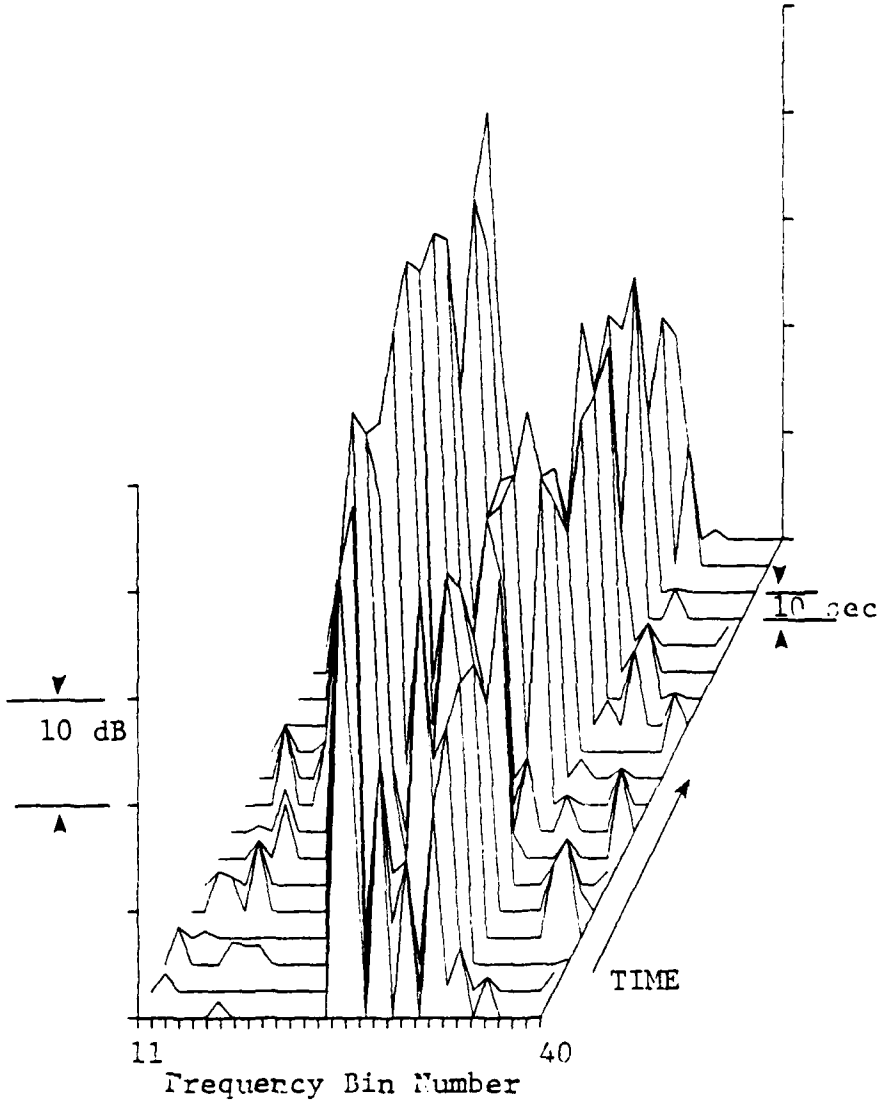
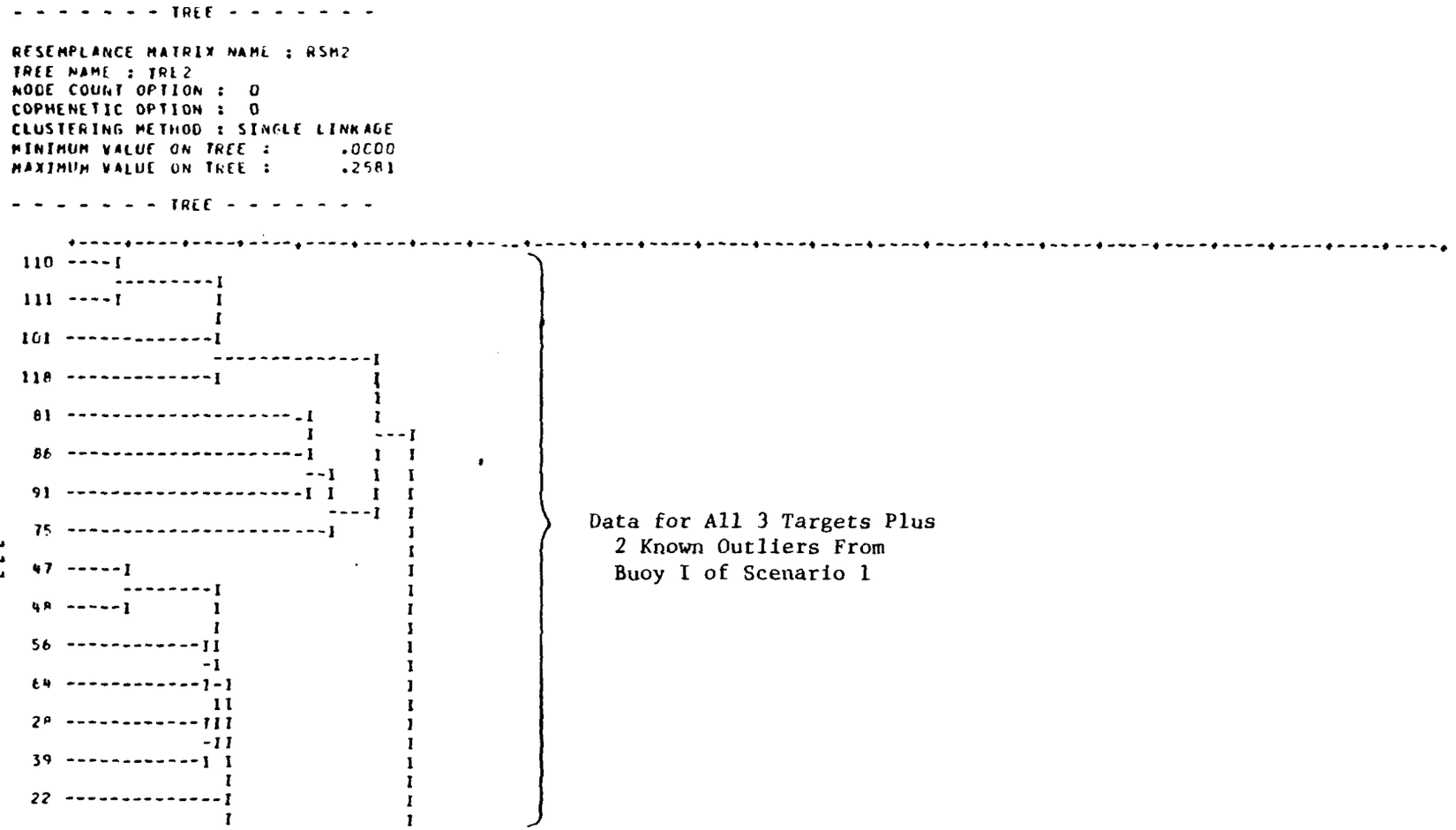
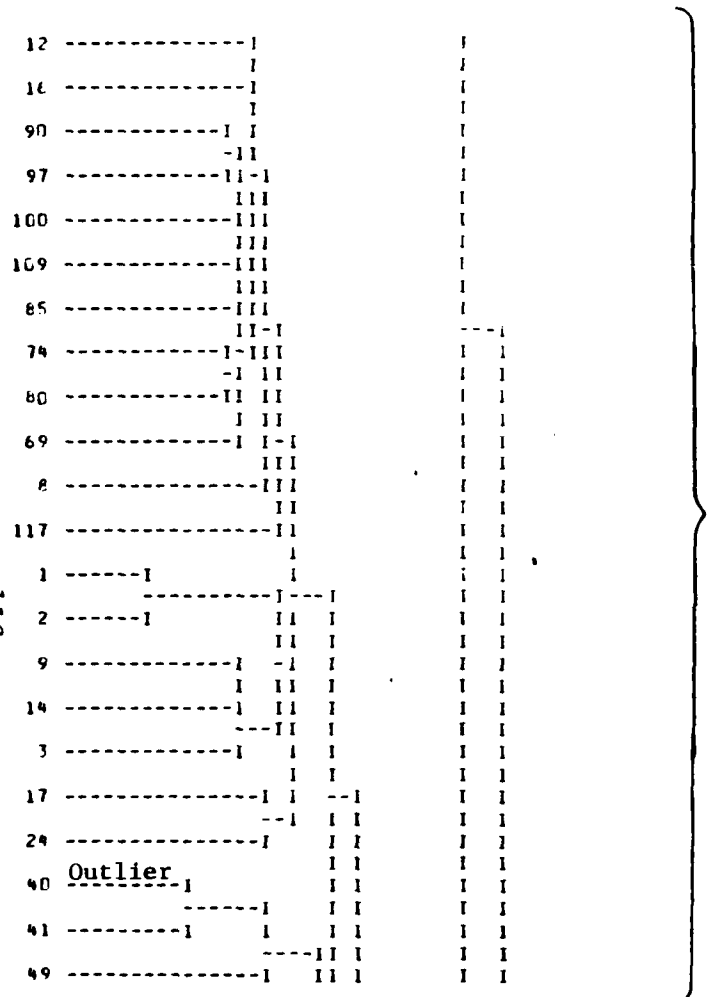


Figure 5.5 - PLOT OF SIMULATED MULTIPLE TARGET FREQUENCY SPECTRA FOR BUOY I OF SCENARIO 1 (3 TARGETS)



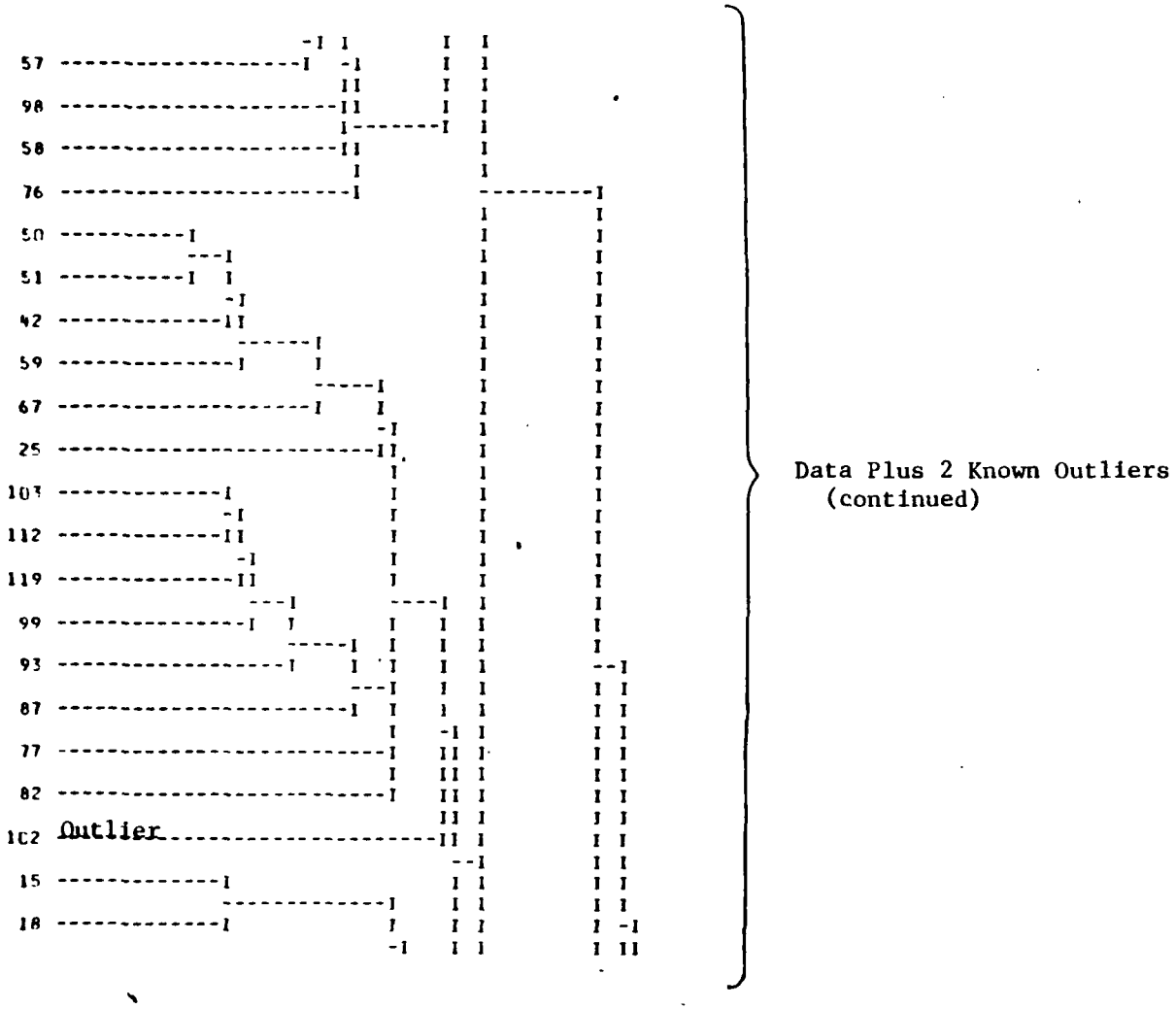
111

Figure 5.6 - CLUSTERING TREE DIAGRAM FOR MULTIPLE TARGET FREQUENCY SPECTRA FROM BUOY I OF SCENARIO 1



Data Plus 2 Known Outliers
(continued)

Figure 5.6 -- Continued



Data Plus 2 Known Outliers
(continued)

Figure 5.6 -- Continued

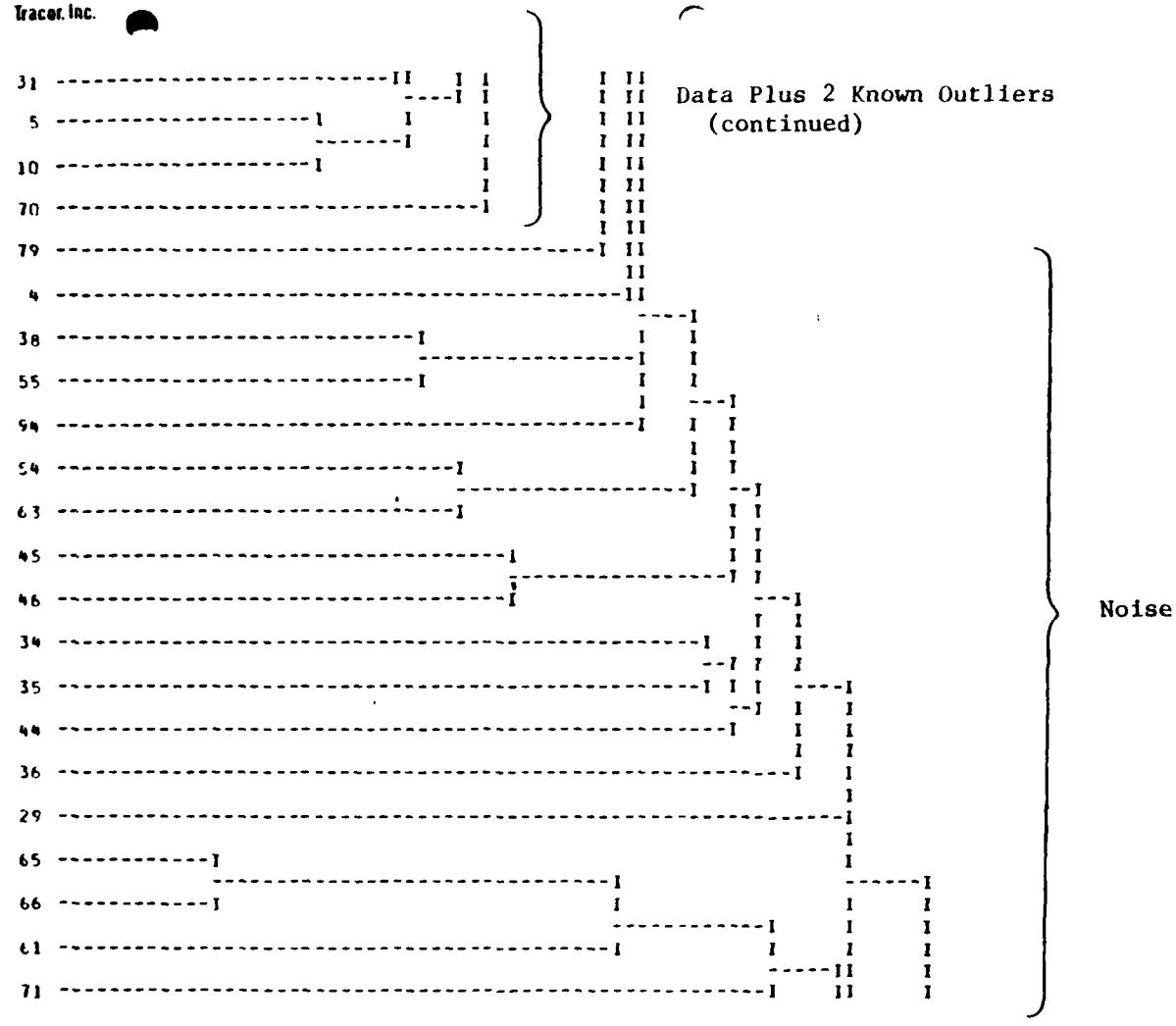


Figure 5.6 -- Continued

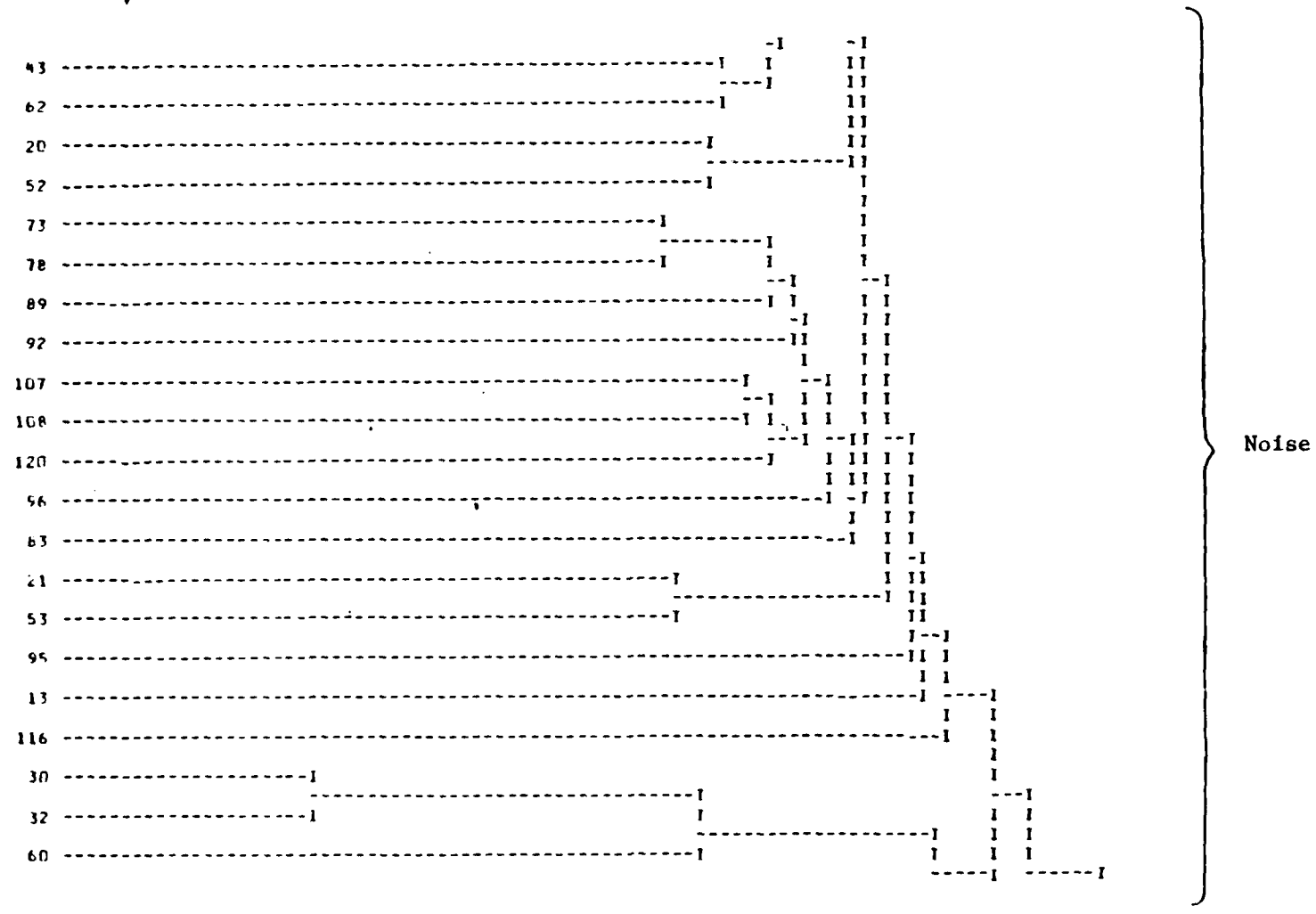


Figure 5.6 -- Continued

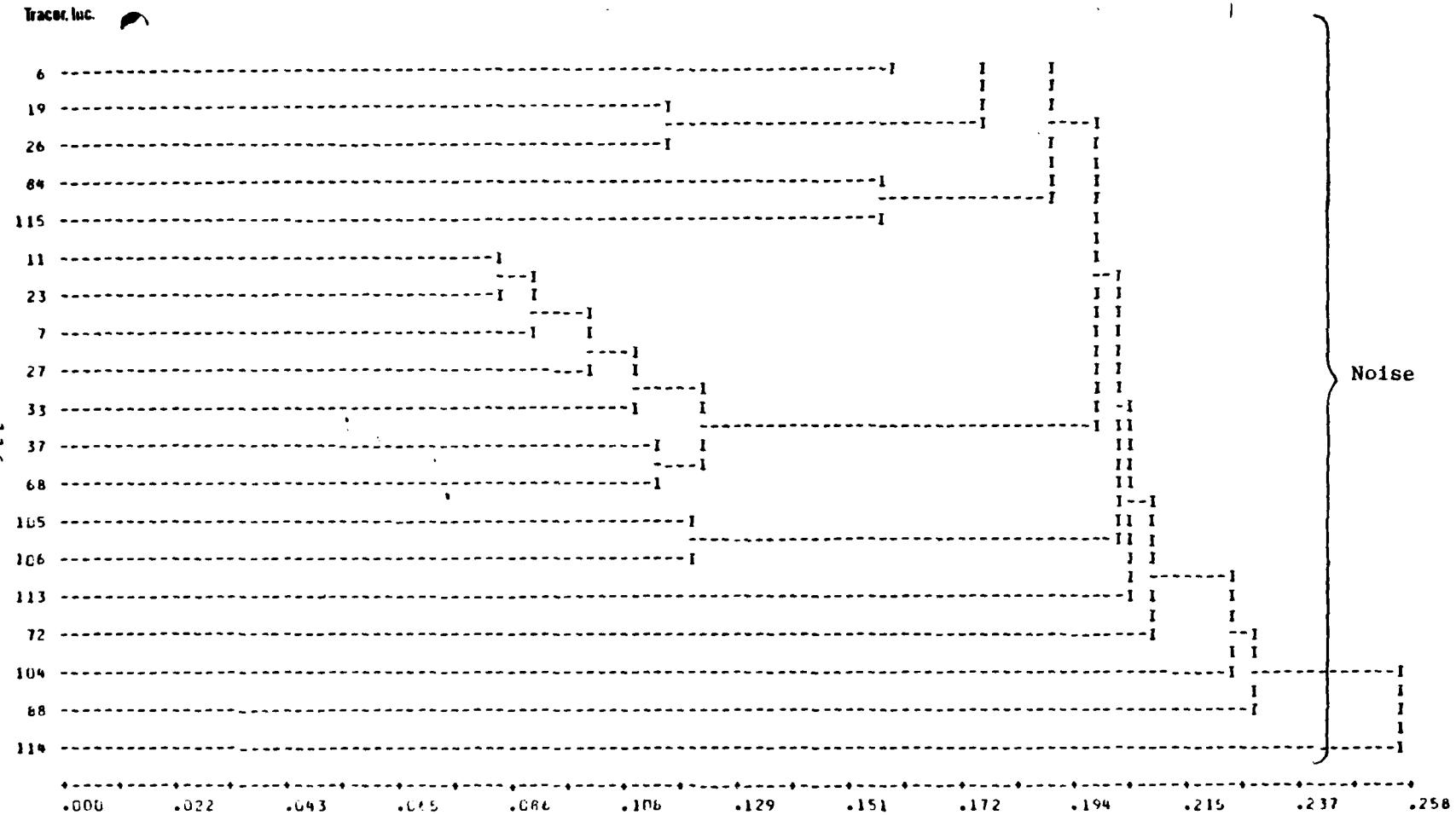


Figure 5.6 -- Concluded

110 and 75 in this tree diagram consist of a partial data set for target 1. Next, from samples 47 through 2 are found most of the data for target 1. However, the last two samples of this cluster, samples 1 and 2, are shown really to be more tightly associated with the next cluster of data which contains the remainder of the data for target 2. The cluster of data between samples 9 and 76 contain the remainder of the data for target 2 except for outlier sample 40. The remainder of the tree diagram that contains the real data, samples 50 through 70, has the signal data for target 3. Again, sample 102 should be excluded from this last cluster because it really is an outlier. All of the true measurement data are found in the upper portion of the tree diagram up through sample 70, but the previous discussion has shown that cluster analysis only succeeded in separating the signals from the noise. It did not properly sort the data into individual sets for each target. To properly separate the signal data from the noise, a threshold value of approximately 0.075 is needed.

The clustering results from sorting the simulated multiple target frequency spectra for sonobuoys II and III of scenario 1 yielded very similar results to those seen in Figure 5.6. For both sonobuoys, the upper half of the clustering tree diagram contained the real data for the three targets. The other data samples were found to be random noise. In neither case were the true data properly sorted into individual target data sets, but the true measurements for all the targets were properly sorted from most of the noise. For sonobuoy II, a dissimilarity coefficient cutoff value of approximately 0.104 would result in all of the true data being separated from all of the noise except for seven outliers which appear in the data sorting. If a threshold value of 0.090 was set for the clustering tree diagram for sonobuoy III of scenario 1, the

resulting data set would contain all the true measurements for the three targets plus eight known outliers. Again, the data were not properly sorted into individual target sets, but the single linkage clustering algorithm did sort the true measurements from most of the noise.

5.8.2 Results from Clustering Multiple Target Frequency Spectra for Scenario 2 - Finally, the single linkage clustering algorithm was tested with simulated, two-target frequency spectra data from scenario 2. For this simulation, a 0.2 Hz difference in the transmitted center frequencies was used to insure that there would be no overlap between the two signals in one frequency bin. As was the case for the data from scenario 1, the clustering approach was fairly successful in separating the true signals from the ambient noise, but it did not adequately separate the data into individual target sets. The results for the three sensors from this scenario are described in the following paragraph.

The clustering tree diagrams for each of the three sonobuoys of scenario 2 again sorted the simulated frequency spectral data so that all of the true measurements were found in roughly the upper half and the noise in the bottom half of the diagrams. However, in no case were the true measurements properly sorted into individual target data sets. Also, the tree diagrams associated with sensors II and III included at least a few outliers in the separated measurement set. Only sensor I completely eliminated any obvious outliers when a dissimilarity coefficient value of approximately 0.060 was used to separate the true data from the noise. For sensor II of scenario 2, a threshold level of 0.065 for the dissimilarity would separate the true data from most of the noise, but would result in six known outliers showing up in the

measurements. Finally, a cutoff value of 0.050 would be needed for the dissimilarity coefficient for sonobuoy III to separate the true measurements plus two outliers from the remaining random noise. Once again, the measurements were not properly sorted into individual target sets.

5.8.3 Conclusions from Using the Single Linkage Algorithm to Sort Simulated Frequency Spectra Data - The results from applying the single linkage clustering approach to data sorting at the frequency spectra level have been both encouraging and discouraging. The encouraging results have been that this approach can separate the multiple narrowband frequencies from most of the ambient noise found in the frequency spectra from the simulated DIFAR processor. One of the discouraging results is that this approach does not suitably sort the data into individual target sets. Another discouraging result is that once again, no hard fast rule for adopting a clustering threshold level can be readily chosen by reviewing the results of these studies. For scenario 1, the threshold levels varied from 0.075 to 0.104. For scenario 2, these levels varied from 0.047 to 0.065. Some of the discrepancies in clustering threshold may be caused by the data normalization method employed. Perhaps this threshold level is a dynamic factor which must be allowed to vary from one problem to another. The question is what type of dynamic relationship can be assigned to the program or what type of normalization scheme should be used to allow the results to become automated rather than depending on human interpretations to pick the optimal clusters. Despite the shortcomings of this approach, it now appears as though this technique can be used to pick multiple peaks from a DIFAR processor so that data sets of the form used in Subsection 5.7 could be gathered for multiple target scenarios. Should this be the case, then the results

from Subsection 5.7 would indicate that the data could again be clustered to sort out outliers and to separate the data into sets of individual target data. These results would seem to suggest the need for a two stage clustering scheme. The first stage would separate the multiple target signals from the noise and the second stage would sort the data into individual target sets.

5.9 Conclusions on the Use of Cluster Analysis for Data Sorting in the Multiple Target Problem

Generally speaking, the hierarchical, non-overlapping single linkage clustering algorithm chosen for this study has shown potential for solving the data sorting problem associated with multiple target tracking. The single linkage cluster analysis program has been used to investigate three facets of the data sorting problem. One study investigated the use of cluster analysis to solve the outlier removal problem. Another study was concerned with the question of sorting multiple linetracker target data into individual target data sets. The final investigation concerned the use of this single linkage clustering algorithm to sort multiple signals from ambient noise found in frequency spectra data. Qualified success has been found in using the cluster analysis approach to solve these problems.

The major problem associated with the clustering algorithm concerns automating the program to pick the optimal set of clusters and to output these results in more useful formats than the tree diagrams found in this report. As the tree diagrams have shown, the clustering algorithm continues to link the data until all points are joined into one conglomerate cluster. The useful information to be gathered from the

clustering tree diagrams falls at intermediate clustering levels rather than at the final level. No fixed criteria have yet been devised to automatically decide when the linking of the clusters should be stopped. Since simulated data were used in these studies, the optimal results were known a priori and an appropriate clustering threshold level could be found. With real data, this will not be possible. In general, the following two observations can be made concerning the choosing of optimal clustering levels. First, the good data were always found in the upper portion of the tree diagram with most outliers or random noise points being found at the bottom of these diagrams. This is true because these trees are arranged in order of increasing dissimilarity coefficients. Secondly, the optimal clusters containing the true data were usually much more tightly knit than the clusters which either joined data from other targets or which included outliers into the cluster. Perhaps some scheme can be devised which gradually picks successively lower clustering levels in the tree diagram until some optimal clusters are found. Great emphasis, especially for separating signals from noise from the frequency spectra and for outlier removal problems, should be placed on analyzing the upper portion of the tree diagram. Another possible improvement would be to standardize the data normalization approach so that all of the raw data from each buoy were normalized by the same scale factor. With such a common scale factor, it may be possible to establish a fixed threshold level for the clustering results. Regardless of how it is accomplished, some criteria must still be developed which determines how much of the upper portion of the tree diagram should be analyzed and how this cluster should be further sorted into clusters of individual target data.

In reviewing the results of sections 5.6 through 5.8, it seems that the input data processing for the multiple target problem should employ the following approach for DIFAR data. First, a rather wide frequency band should be chosen for observation which includes all of the possible narrowband signals of interest. Next, some threshold test should be employed which accepts most of the signal data plus some noise data but which rejects most of the random ambient noise. For all data that pass the threshold test, a 4-tuple of attributes should be estimated which includes the time tag, the frequency estimate or the frequency cell bin number, and the sine and cosine of the bearing estimate. This set of attributes for all the prospective signal data should then be analyzed by a two stage clustering algorithm. The first stage of this clustering phase would be used to separate the signal data from most of the remaining random ambient noise found in the frequency spectra. Assuming that the output from the clustering algorithm has been suitably automated, the resulting signal data would be separated from the noise and saved for another round of clustering. The second clustering would serve two purposes. First, it should remove the remaining outliers from the signal data. Second, the algorithm should decide how many targets are present and assign optimal clusters of data to each of the targets believed to be present. It is felt that some criteria can be developed to automate the clustering algorithm to perform these tasks, but as yet, no obvious method has been found. Perhaps with a proper data normalization scheme, some of the problems concerning the automation of cluster analysis output can be more easily solved. With futher development, it is strongly felt that cluster analysis can be used to identify data, determine how many targets are present, sort the data into individual target sets and eliminate any outliers that do not truly belong in a given data set.

6.0 RECOMMENDATIONS FOR FUTURE INVESTIGATIONS

Upon completion of the current contract, it is evident that considerable work remains to be done in the area of multiple target tracking. In connection with this, several future tasks have been identified. The first two of these tasks are concerned with improving the capability to sort multiple target data. Other tasks are concerned with utilizing the sorted data properly to track the multiple targets described by the data. These proposed tasks are generally presented in the required order for their logical development. A few concluding comments address long term work on multiple target tracking.

6.1 Continued Search for the Optimal Clustering Technique

First, it is felt that the search for the optimal clustering algorithm must be continued. This initial study has shown that CLUSTAR's single linkage algorithm and Ling's (1,r) algorithm were the best of the algorithms tested for the data sorting problem associated with multiple target data. Both of these algorithms are hierarchical, non-overlapping, single linkage clustering algorithms. From the results of this study and from heuristic reasoning, it is believed that either a more generalized, hierarchical and non-overlapping single linkage algorithm or an overlapping, non-hierarchical single linkage algorithm may be better suited for the data sorting problem. A brief discussion of these ideas follows.

One possible investigation on this topic concerns the development of Ling's generalized (k,r) clustering algorithm for $k \neq 1$. Ling points out (Reference 11) that this

algorithm should best be thought of as a generalization of the conventional single linkage algorithms. The k and r control parameters are used to determine the level to which objects or clusters should be linked by the algorithm. This (k, r) algorithm merges data into groups of k members that are all linked within some distance r of the other members in the group. Both k and r may be user inputs that would be used to determine how much the data should be linked before the clustering process would be stopped. It is believed that the (k, r) algorithm is a necessary generalization of the single linkage algorithm which should be easier to control and automate and which could prove to be more useful for the data sorting problem than the conventional single linkage algorithm.

A second clustering algorithm which should be investigated is a non-hierarchical, overlapping algorithm. This algorithm is referred to as the Moody and Jardine B_k algorithm (Reference 8). It too is a generalization of the single linkage algorithm. For this algorithm, the k parameter is used to define the degree of overlap that is to be allowed between two different clusters. For an overlapping clustering algorithm such as this one, data are not always assigned to only one cluster. Instead, data that cannot be clearly separated into either cluster are placed into both and the clusters are allowed to overlap at this point. This type of algorithm may prove useful for pathological cases such as that found in scenario 2 of this study when the data from two targets are so similar that they cannot be readily separated. Rather than assign the questionable data points to one target or the other, it may prove to be more useful to assign these points to both targets. This approach could be especially useful in situations where target trajectories intersect or nearly intersect. It may even be easier to automate this algorithm than

the currently used hierarchical, non-overlapping algorithm, but it is difficult to speculate until the algorithm has been built and tested. Regardless, this Moody and Jardine B_k algorithm is another approach to generalizing the conventional single linkage algorithm which is felt to have potential and therefore deserves consideration for future studies.

6.2 Automating the Multi-Target Clustering Algorithm

The best multi-target cluster algorithm determined from the previous task must be automated before it can be used in any practical system. If the previous two algorithms are developed and the results prove to be unsatisfactory, then it will become necessary to attempt to automate the output of the existing single linkage algorithm. In this context, automating means that the clustering algorithm will be modified to decide for itself what the optimal number of clusters are and how the data should be assigned to these clusters. A study of attribute normalization is an essential feature of this task. The clustering algorithm would also be modified to output the data in a tabular form rather than in tree diagrams. Finally, the samples for each cluster should be automatically reordered to appear chronologically correct so that a tracking algorithm could properly process the data. Some of the modifications will require changes to the existing clustering packages and others may require the development of some post processing programs for the sorted data. Nevertheless, if cluster analysis is ever to be successfully used to sort data for target tracking problems, the clustering algorithms must be automated to output the data in the form needed by a tracking algorithm.

6.3 Intersensor Data Matching Procedure

After some type of data sorting procedure has been adopted, data from different sensors for the same targets must be properly matched so that tracking solutions may be obtained for all of the targets. With passive DIFAR data, only frequency and bearing estimates are generated for each signal. Previous studies have shown that frequency and bearing measurements from only one sensor are usually insufficient for initializing or tracking unless the data spans a considerable range in frequency and bearing. When only passive frequencies and bearings are to be used for target tracking, one should have overlapping measurements from at least two sensors to insure accurate, timely tracking results. For multiple target problems, the question then becomes how to match the individual target data sets from one sensor with those from another sensor. One could simply use a trial and error scheme for matching the data sets until reasonable solutions were found, but some more organized and quicker scheme for doing this would be preferred. There are several suggestions for solving this problem.

One possibility would be to use the initial guess procedure described in Section 2 to pick the most likely pairings and to eliminate the impossible pairings of data sets from two or more different sensors. If a reasonably accurate initial guess is used, it would be possible to pair data sets together to estimate an initial position and velocity guess for that pair. Physical constraints on the range of the detection systems and on the allowable velocities for ships could be used to immediately eliminate impossible pairings of the data sets. After the impossible pairings have been eliminated, the target tracking algorithm could be initialized with the allowable

guesses passed by the tracker's initial guess procedure. After processing some more data, better estimates for positions, velocities and accelerations from potential data pairings could be found. Once again, physical constraints could be used to eliminate the impossible solutions produced by certain pairs of data sets. Basically then, a good initial guess procedure would be more useful for eliminating impossible pairings of data sets than for picking the most likely pairings of the individual data sets. Nonetheless, such a process would be extremely valuable in reducing the complexity of the data matching problem.

A second approach would be to use the 0-1 integer programming techniques as proposed by Morefield (Reference 13). This approach would require the development of a cost function which would be minimized by picking the correct pairing of data sets for individual targets. The set of pairings which minimizes this cost function would be chosen as the proper pairings of data from that data set. The idea with this approach would be to pick the most likely pairings of the data out of all the possible combinations that could be generated by a given buoy pair.

Another appealing approach would be to combine the two techniques described above into a joint intersensor data matching scheme. First, the initial guess procedure would be used in conjunction with the physical constraints on the targets and the measuring devices to eliminate the impossible pairings of data sets. This could still leave possible pairings of data sets from the different sensors that would exceed the number of targets thought to be present in the observation range. Next, the 0-1 integer programming approach could be used to analyze the possible pairings and pick the most likely set from these allowable pairs.

This latter approach seems to be a very appealing one to pursue. The initial guess and elimination procedure should be fairly easy to implement and should prove to be reasonably quick with regard to computer applications. The integer programming approach will be more difficult to develop and implement and will probably require substantial amounts of computer memory and time to pick the optimal sets of data pairing. If speed and memory limitations are to be important considerations for this intersensor data matching problem, then the integer programming technique should be used only when necessary. If pairings can be eliminated before the data sets are passed to the integer programming algorithm, substantial savings in computer time and memory should result. Thus, in the interest of simplifying the decision making process and of speeding up this process, both of these approaches should be merged to create an intersensor data matching procedure.

6.4 Other Problems

The three subjects discussed above are planned areas of work for the near term. The following subsections mention several other areas identified as requiring work. It also discusses some major research topics in multi-target tracking which are currently deferred to later investigations.

6.4.1 Identification of Redundant Data Sets - Another topic which should be investigated concerns the identification of multiple or redundant data sets for the same target as observed by one sensor. If one observes a broad frequency band to detect multiple narrowband signals, it becomes likely that multiple lines from only one target will be observed in the data. One cause for multiple signals would be the presence of

harmonic multiples of a fundamental frequency from a given target. When multiple signals from a single target occur, one would like to be able to identify the multiple lines and group them together for each of the targets. Being able to identify redundant lines would be extremely valuable for determining how many targets are actually present in a given set of frequency spectra. Furthermore, if these multiple lines were identified, one could pick only one line to be used for each target in the intersensor data matching problem and substantially reduce the number of possible combinations that need to be examined by this processor. Thus, some sort of scheme for identifying redundant lines from one target would be valuable in determining the total number of targets found in a given set of frequency spectra and in reducing the complexity of the problem to be solved by the intersensor data matching processor.

6.4.2 Compensation for Data Dropout - Another task that warrants investigation concerns the compensation for data dropout that arises when acoustic data is gathered. A variety of factors can lead to signal fading or possibly to periods of data loss. These factors included propagation losses, smearing losses, and random fluctuations in signal strength and ambient noise levels. Some of the trial results not discussed in this report showed that problems can be encountered with our data sorting scheme when a signal temporarily fades from view. If only one, two or maybe even three consecutive measurement updates are lost, the cluster analysis algorithm could successfully continue to sort the data into correct data sets. However, in cases where four or more consecutive update times were encountered with no measurement output, the clustering algorithm improperly sorted the data after the signal was recovered. Rather than joining the data from before the temporary data loss period to the data recovered after this

loss, the clustering algorithm output two distinct clusters which one would associate with two different targets. Clearly the loss of data over this interval leads to data sorting problems.

Two possible solutions to this problem can be offered. One approach would be to prefilter the data to fit some kind of curve or surface to the data. When no measurements were output for given time points, one could use the fitted curve to predict what a measurement value should have been and use this value as a substitute for the missing measurement update. Another approach would be to use a target tracking algorithm to predict what the measurement value should have been. Provided the tracker has converged onto a legitimate solution, the best estimates from the last time point can be integrated forward to predict what the next measurement should be. This predicted measurement could be used to replace lost data when data dropout occurs. Either of these two approaches will probably succeed, but only when the target moves along a non-maneuvering, constant velocity trajectory. If the target is involved in some kind of maneuver when data loss occurs, neither of these prediction schemes are likely to compute good estimates for the missing data points. These two prediction schemes seem to be the most likely techniques to be used to compensate for data loss, but both have some pitfalls. It will not be known how effective either approach can be without experimenting with some data sets and then analyzing the results to determine the efficacy of these approaches.

6.4.3 Long Range Research Topics - Many other problems exist which will need to be investigated before a robust multiple target algorithm can be built. For instance, it is

still not known if the matched multiple target data should be processed by a bank of single target tracking algorithms operating in parallel or if a new multiple target tracking algorithm must be developed which updates all of the individual tracks simultaneously. Still another task that merits research would be the development of an a posteriori processor to determine if the data have been properly sorted by the clustering process and then correctly matched with the intersensor data matching processor. It seems that an a posteriori processor could be merged with the cluster analysis and intersensor data matching processor to develop a predictor-corrector type of approach to the sorting problem associated with the multiple target tracking problem.

Besides the problems mentioned, still other questions are likely to arise as the investigations into the multiple target tracking problem continue. However, the tasks proposed here are believed to be a natural progression to the work begun and described in this report. Qualified success has been attained in our initial data sorting study and the proposed tasks should lead to further progress on this difficult problem. Solutions from these tasks should lead to the successful implementation of future target tracking systems. For the present, however, it still remains to be seen if a fully automated system can be developed to track multiple targets even within constrained scenarios.

7.0 REFERENCES

1. Corser, Glenn and Wilson, Thomas, Final Report, Hybrid Passive Tracking Algorithms, Contract N00014-78-C-0670, 31 October 1980.

2. Maybeck, Peter S., Stochastic Models, Estimation and Control, Vol. 1, Academic Press, Inc., New York, 1979.

3. Escobal, Pedro Ramon, Methods of Orbit Determination, Robert E. Krieger Publishing Company, New York, 1976.

4. Reeder, Hugh, Final Report, A Maximum Likelihood Procedure for Air ASW Program (U), Contract N60921-79-C-0123, Tracor, Inc., Report T80-AU-69-C, 15 May 1980.

5. Burden, Richard L., and Faires, J. Douglas, and Reynolds, Albert C., Numerical Analysis, Prindle, Weber and Schmidt, Boston, 1978.

6. Urick, Robert J., Principles of Underwater Sound for Engineers, McGraw-Hill, Inc., New York, 1975.

7. Sneath, Peter H. A. and Sokal, Robert R., Numerical Taxonomy, W. H. Freemand and Company, San Francisco, 1973.

8. Jardine, Nicholas and Sibson, Robin, Mathematical Taxonomy, John Wiley and Sons, Ltd., London, 1971.

9. Anderberg, Michael R., Cluster Analysis for Applications, Academic Press, Inc., New York, 1973.

10. Marshall, Kim and Romesburg, Charles H., "CLUSTAR and CLUSTID - Programs for Hierarchical Cluster Analysis," The American Statistician, Vol. 34, No. 3, August, 1980.

11. Ling, Robert F., "On the Theory and Construction of k-Clusters," The Computer Journal, Vol. 15, No. 4, November, 1972, pp. 326-332.

12. Ling, Robert F., "A Probability Theory of Cluster Analysis," J. Am. Stat. Assoc., Vol. 68, No. 341, March, 1973, pp. 159-164.

13. Morefield, C. L., "Application of 0-1 Integer Programming to Multitarget Tracking Problems," IEEE Trans. Automatic Control, Vol. AC-22, pp. 302-311, June, 1977.

APPENDIX A

RESPONSE SURFACE METHODOLOGY (RSM) STUDY OF THE HYBRID TRACKING ALGORITHM

A.0 INTRODUCTION

A significant task under this contract was to analyze the performance and parameter dependence of the Hybrid tracking algorithm. Three control variables were used in this study. They were signal-to-noise ratio (SNR), data integration time (INT), which coincided with the time between successive measurement updates for this study, and buoy separation distance (SEP). Performance of the algorithm was measured by the following variables: the average distance error for the tracking solutions (ADE), the predicted distance error (PDE), i.e., the difference between actual and predicted position 300 seconds after the end of data acquisition, and convergence time (CT), the time required for the batch initializer to converge to a trajectory that is within 500 meters of the true trajectory.

To perform the analysis of Hybrid's performance, a statistical technique known as Response Surface Methodology (RSM) was used.* RSM essentially uses multiple regression to relate the response of a particular system or process to the various inputs (independent variables) which are assumed to affect it. The goal of RSM is to create a surface which accurately reflects the system response function and then explore this surface for extrema and optimal operating areas. RSM is closely related to the field of experimental design and most RSM plans or designs have their origins in the ideas developed by statisticians working in the areas of analysis of variance and statistical design of experiments.

---

* Myers, Raymond H., Response Surface Methodology, Allyn and Bacon, Inc., Boston, 1971.

A.1 Description of Independent Factors

In order to design the RSM experiment, decisions had to be made on which factors should be used to investigate the Hybrid's tracking performance. It was decided that of the possible factors, data quality most affected the Hybrid's tracking solutions; so the three factors which govern the data quality were chosen for the independent RSM parameters. The three independent parameters chosen were the separation distance between sensors (SEP) in a triangular sonobuoy deployment pattern referred to as tri-tac pattern, the data integration time used to gather the data (INT), and the signal-to-noise ratio of the transmitted signal (SNR). For this study the integration time coincides with the data update rate and the signal-to-noise ratio is the usual difference in the source level and the ambient noise level in dB.

A.1.1 Sensor Separation Distance (SEP) - SEP was chosen as an independent factor because the quality of the tracking was strongly influenced by the placement of the sonobuoy pattern used to observe the target. For all of the test design points, an equilateral tri-tac pattern was used with sonobuoys placed at each of the three vertices. The distance between sensors was varied for each design point in such a fashion as to allow the centroid of the pattern to remain fixed. In actual practice, operators have some control over where sonobuoys are initially deployed. Once the sonobuoys are dropped, however, their motion is governed only by the ocean currents and winds. For this study, it was assumed that the sonobuoys were dropped onto precisely known positions and remained stationary throughout the scenario. From the test design, Hybrid's response to the positioning of the sonobuoys and to their relative separation could be determined and analyzed so that optimal separation distances could be found.

A.1.2 Integration Time (INT) - The second independent factor chosen was the data integration time (INT). For the data simulation program used, it was assumed that the measurement update equation was governed by

$$\beta\tau = 1$$

where β is the resolution of the frequency measurement in Hz and τ is the integration time in seconds that is used to generate the estimate. For this study, it was assumed that INT coincided with the update intervals for generating the frequency and bearing estimates. Reviewing this equation, it can be seen that the resolution of the frequency estimate is inversely proportional to INT. When INT is small, the corresponding resolution of the frequency estimate will be very coarse due to this inverse relationship. Conversely, to obtain frequency estimates with a very fine resolution, large values for INT must be used. This equation for relating resolution of the frequency estimates to the integration time period chosen is true for most passive acoustic detection systems that are used. With this model, the trade-off between accuracy in frequency estimates and data update intervals could be examined to determine its effect on the tracking response of the Hybrid algorithm. INT is a factor that an operator can control and vary with time, so this RSM study will determine what values for INT should be chosen to optimize Hybrid's tracking response.

A.1.3 Signal-to-Noise Ratio (SNR) - The last independent factor chosen was the signal-to-noise ratio (SNR) of the transmitted signal as measured one yard from the target. This factor is completely removed from control of the operator. It is only a function of the target's transmitted

signal strength and the ambient noise level of the ocean. The transmitted signal strength varies from target to target and the ambient noise level varies according to the sea state of the environment. SNR, as detected at a sonobuoy receiver, is also a function of the range from the target to the individual buoy. The propagation loss for the signal passing through the water is assumed to be 20 log (R) in dB, where R is the magnitude of distance from the target to the sonobuoy's receiver. The propagation loss is idealistic, but over the ranges and depths involved, it is a reasonable approximation for the purpose of this study. As will be detailed in subsequent sections, other random fluctuations in signal strength and ambient noise level are also assumed to influence the computed value of SNR at the receiver. Basically, however, SNR is a function of the signal strength, the ambient noise level, and the distance between the target and receiver. The only controlling factor an operator would have on SNR would be to deploy the sonobuoys very close to the target, but generally, the target position will not be known very accurately a priori. This may then be thought of as an uncontrollable factor.

A.2 Response (Dependent) Factors for RSM Study

Three different dependent factors were used to define Hybrid's response (i.e., performance) at the various design points. The three responses used were the same as those used previously (Reference 1) to quantify Hybrid's tracking capabilities. Separate response surfaces were generated for each of the response factors. The three dependent, or response, factors used were the average distance error of the estimates (ADE), the time where the Hybrid successfully converged upon a satisfactory set of initial conditions (CT),

and the distance error incurred by predicting the final estimate forward five minutes after the last data point was processed (PDE).

A.2.1 Average Distance Error (ADE) - ADE is one measure commonly used to describe the accuracy of an estimated target solution output by a target tracking algorithm. This measure provides an indicator of how well tracker estimates fit the actual trajectory over a portion of the trajectory where there is data. ADE is defined as:

$$ADE = \left(\frac{1}{t_f - t_o}\right) \int_{t_o}^{t_f} \sqrt{(\hat{x} - x_T)^2 + (\hat{y} - y_T)^2}\, dt,$$

where ($\hat{}$) denotes the estimated solution at time t and the subscript T denotes the true value at time t. ADE then provides an average of the position error between the estimated and the true target trajectory over the entire length of the accumulated data stream.

A.2.2 Convergence Time (CT) - Unfortunately, ADE by itself does not always provide a sufficient measure of the tracking performance for a given tracker such as Hybrid. One not only wants an algorithm that yields minimum ADE, but also an algorithm which converges as rapidly as possible onto a suitable estimate for the target's trajectory. For the Hybrid in particular, quick convergence is preferred because the tracker will switch from the computationally slower batch initializer to the faster sequential tracker as soon as its convergence criteria have been met. For this study, the convergence time was chosen to be the time at which the tracker switched from the batch to sequential filter and the tracking

results eventually yielded position estimates that fell within 500 meters of the true position values. In cases where these criteria were not met, a time which corresponded to the end of the scenario was assigned to CT.

A.2.3 Predicted Distance Error (PDE) - The last factor used to measure Hybrid's tracking response was PDE. This measure is used to determine the Hybrid's capability for prediciting a target's position five minutes after the last data point has been processed. In general, tracking algorithms, such as Hybrid, which use a suboptimal motion model will often yield satisfactory results for CT and ADE, but will prove to be a poor predictor. The predictive capabilities of a tracker are of interest for weapons and sensor deployment. This third factor combines position and velocity errors into a single measure. From this study, control values are sought to *maximize Hybrid's predictive capabilities.*

A.3 Orthogonal Central Composite Design (OCCD)

This RSM study, uses an orthogonal central composite design (OCCD) to select the design points used to generate the quadratic fits for the response surfaces. The independent factors SNR, SEP, and INT were varied according to this experimental design so that the quadratic response surface could be obtained with a minimum number of design points. The OCCD is essentially a 2^3 factorial design augmented by a center point and axial points that are chosen so as to produce zero correlation among all factors and, derivatively, their coefficients. Geometrically, this design consists of a cube with experiments being performed at the corners, the center point, and the ends of axial lines passing through the center and perpendicular to each cube face. Practically, this means

that computer simulation runs were made with the SNR, SEP, and INT values given at each of these design points. The values of the three responses -- ADE, PDE, and CT -- were found at each point and a quadratic surface relating each response to the independent variables was constructed.

The OCCD is a standard RSM design. There are several reasons for using it, among them are:

(a) To fit a quadratic surface, all factors, at least, must occur at three levels. However, 3^k experimental designs (k factors each occurring at three levels) contain large numbers of experimental points which are used to estimate higher order interactions and not main effects, quadratic terms, or first order interactions. An OCCD, on the other hand, uses far fewer points and estimates only main effects, quadratic terms, and first order interactions.

(b) Since each factor in the OCCD occurs at five levels, this design offers broader actual experimental coverage of the area of interest.

(c) The axial points can be chosen so that the correlation between all the estimated parameters is zero.

It can be seen that an OCCD presents an ideal design for obtaining quadratic response surfaces. This design uses a minimum of experimental points to obtain a fit, allows

quadratic fits to be made for the data, and results in uncorrelated estimates for the coefficients. Originally, one OCCD was used which was fully expected to test the limits of Hybrid's tracking capabilities. Unfortunately, some of the experimental points yielded poor quality data which prevented the Hybrid from converging onto an accurate solution. Rather than use this poor initial design, it was decided to use a second OCCD which would allow us to accurately model the Hybrid's responses to values of the three independent factors that fell within Hybrid's actual operating range. The first design will be discussed next, along with a description of its shortcomings. Following this, the revised OCCD that was used for this study will be described and will be followed by detailed analyses of the RSM results obtained from this design.

A.4 Description of the Two OCCD's Used

Originally, an OCCD was chosen that was intended to test the limits of the Hybrid's tracking capabilities. Over 20% of the experimental data points failed to yield sufficient data for the Hybrid to track the target. Problems were caused by extremely poor quality data, lengthy periods of data droput, and in some cases, insufficient data for the tracker to be initialized. With such a large void in the data from this design, it was decided that a reasonable, least squares, quadratic fit would not be obtained so no response surfaces were generated for this original design. The following is a description of the design as well as a list of causes for the problems encountered.

A.4.1 Original OCCD - Since this original OCCD was intended to study the extremes of the Hybrid's performance, wide ranges of values for the independent factors were used.

At one end of the test values for each independent factor, very good results were expected, while at the opposite end of the test values, only marginal tracking results were expected. Unfortunately, some of the expected marginal cases turned out to be impossible for the Hybrid to handle. These results emphasized the great care that should be taken in choosing design points, because these points should provide useful information about the actual operating range of the Hybrid tracking algorithm.

The independent factors used for this RSM study were SEP, SNR, and INT. The values for each of these factors at each design level are given in Table A.I. The corresponding target scenario used for this study, along with the buoy positions for each design level are provided in Figure A.1. Notice that a non-maneuvering target that passes through the buoy field was used for this study. The simulated scenario lasted for 20 minutes. For the SNR values given in Table A.I, the values refer to the signal-to-noise ratio of the signal one yard from the target, not at the sonobuoy's receiver. The propagation loss incurred by the signal passing from the target to the receiver are subtracted from this initial SNR value to compute the SNR at the receiver. These SNR's are given in units of dB. Values for SEP are given in meters. For SEP, the centroid of each buoy pattern used for this study was fixed at $x = 0$ meters and $y = 3500$ meters. The placement of the buoys for each design level was adjusted so as to keep this centroid fixed and to keep the tri-tac pattern in the shape of an equilateral triangle. Finally, the INT values are given in units of seconds. Recall that the resolution of the frequency estimates in Hz is inversely proportional to INT.

TABLE A.I

ORIGINAL OCCD

Factor (z)	Mean (μ)	Delta (Δ)
SNR (dB)	70	8
SEP (m)	7500	2500
INT (sec)	25	20

Scaled axial point for a three factor OCCD:

$$\alpha = 1.216$$

Transformation equation:

$$x = \frac{z - \mu}{\Delta}$$

Experimental Design Values					
z \ x	$-\alpha$	-1	0	+1	$+\alpha$
SNR (dB)	60.272	62	70	78	79.728
SEP (m)	4,460	5,000	7,500	10,000	10,540
INT (sec)	0.68	5.0	25	45	49.32

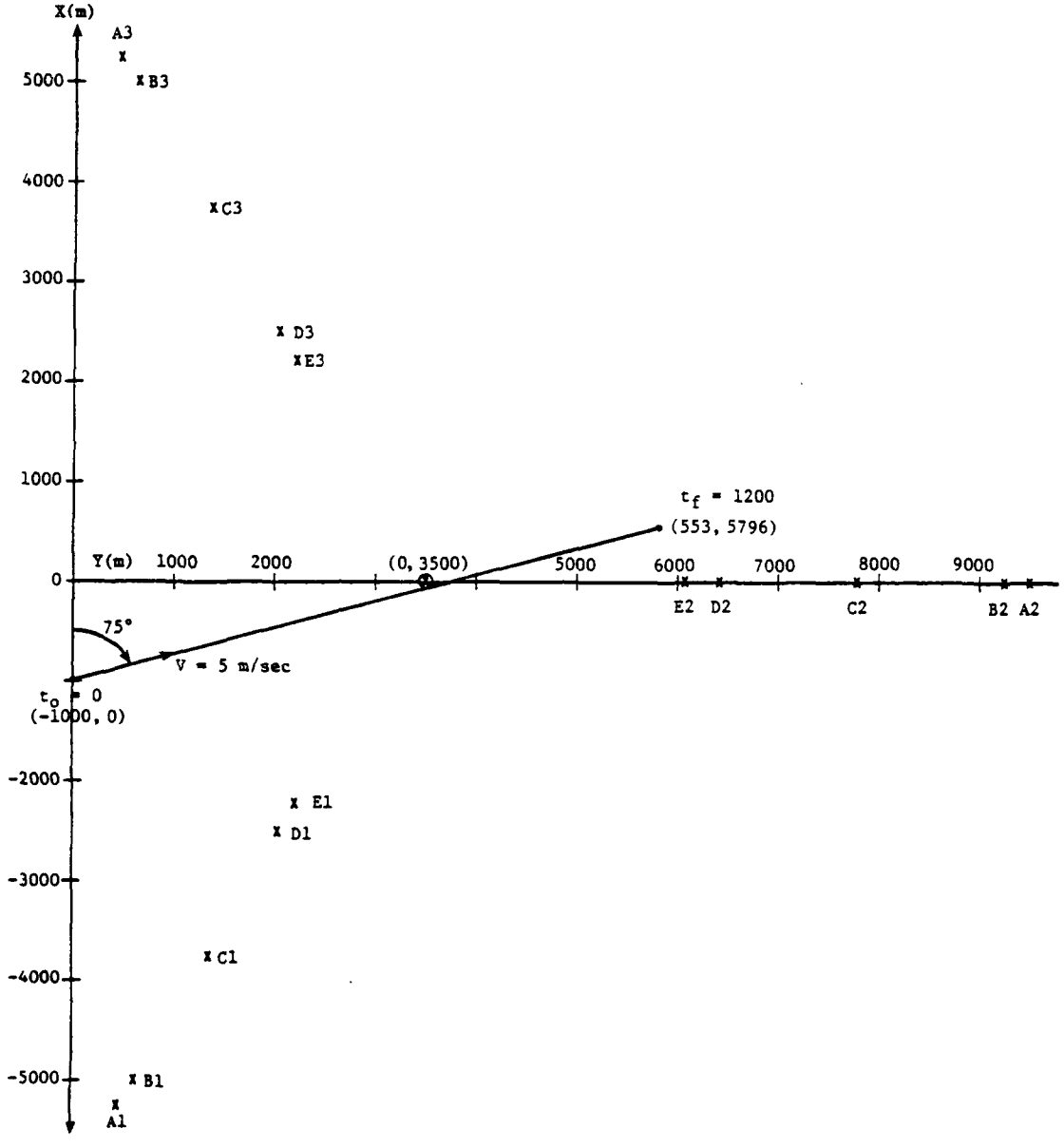


FIGURE A.1 - SCENARIOS FOR ORIGINAL OCCD

Problems were encountered at certain levels for all of these parameters in this initial design. When the values for SEP were greater than or equal to 10,000 meters, insufficient data were obtained for the Hybrid to successfully track the target. The separations were so wide that except for SNR values that would be higher than any of our test values, overlapping data from at least two sonobuoys could not be obtained. Problems were also encountered when the SNR values of 62 and 60.272 dB were used. Again, these values were so small that no overlapping measurements from at least two sensors were found. For the Hybrid to successfully track the target, some interval of overlapping measurements from at least two sensors is preferred to insure quality results. Finally, severe problems were encountered when the smallest value for INT, 0.68 seconds, was used to generate frequency and bearing estimates. Two factors caused this problem. One factor was that the resolution in the frequency estimate was so coarse that little or no change in the frequency estimates was ever seen. Secondly, the integration time is so small and the frequency binwidth is so wide that very little signal is being integrated into the ambient noise for an individual bin. This results in a severely reduced SNR for both the frequency and bearing estimates which severely degrades the accuracy of these estimates. The causes for the problems encountered with this test design may then be summarized as follows:

(1) Two of the sensor separation distances were too large.

(2) Two of the signal-to-noise ratios were too small.

(3) One of the data integration times was much too small.

A.4.2 Revised OCCD - After the problems associated with the original OCCD were carefully studied, a second OCCD was created. This design sought to eliminate the problems previously encountered so that accurate response surfaces for Hybrid's tracking performance could be generated. To revise the test design, the following criteria were used to eliminate the problems previously encountered with the experimental data set.

(1) The range of values for INT were reduced to correspond more closely to rates most commonly used for deployed sonobuoy systems.

(2) The center point of the design values was fixed so as to eliminate very small values for INT.

(3) The range of values for SNR was reduced so that more realistic measurement could be generated by the simulation program.

(4) The mean for the SNR values was increased so that higher overall design values would be used.

(5) The range of values for SEP was reduced so that more overlapping of the individual sonobuoy's observation ranges would occur.

(6) The mean of the values for SEP was also reduced to assure more overlap in measurements from the individual sonobuoys.

(7) The centroid of all the tri-tac sonobuoy patterns was moved closer to the initial starting point of the trajectory to guarantee that stronger signals and more measurements would be available for track initialization.

All of these factors were used to redesign the experimental OCCD. The revised design points are given in Table A.II, followed by a geometric representation of this design in Figure A.2. Figure A.3 shows the scenarios used for this test design, and Table A.III lists scenario parameters.

A.4.2.1 Summary of Results from the Revised OCCD - The results from this OCCD were much improved over those from the first design. The Hybrid was able to converge onto a solution for all the experimental design points. For two of the design points where SEP was large, the Hybrid converged onto a solution, but this solution never converged to less than 500 meters error between the estimated track and the true track. This was probably caused by too little overlap in measurements from at least two sensors, preventing the Hybrid from successfully converging onto the true trajectory. Since the Hybrid never converged to less than 500 meters distance error, a value of 1,200 seconds was assigned to these cases because this time coincides with the final time of the simulated scenario. The results from this revised OCCD experiment are given in Table A.IV. Detailed analyses of the RSM results that were generated for this experimental design are described in detail in the next subsection.

TABLE A.II

REVISED OCCD

Factor (z)	Mean (μ)	Delta (Δ)
SNR (dB)	76	6
SEP (m)	6500	1500
INT (sec)	12.5	7.5

Scaled axial point for a three factor OCCD:

$$\alpha = 1.216$$

Transformation equation:

$$x = \frac{z - \mu}{\Delta}$$

Experimental Design Values					
z \ x	$-\alpha$	-1	0	+1	$+\alpha$
SNR (dB)	68.704	70.0	76.0	82.0	83.296
SEP (m)	4,676	5,000	6,500	8,000	8,324
INT (sec)	3.38	5.0	12.5	20.0	21.62

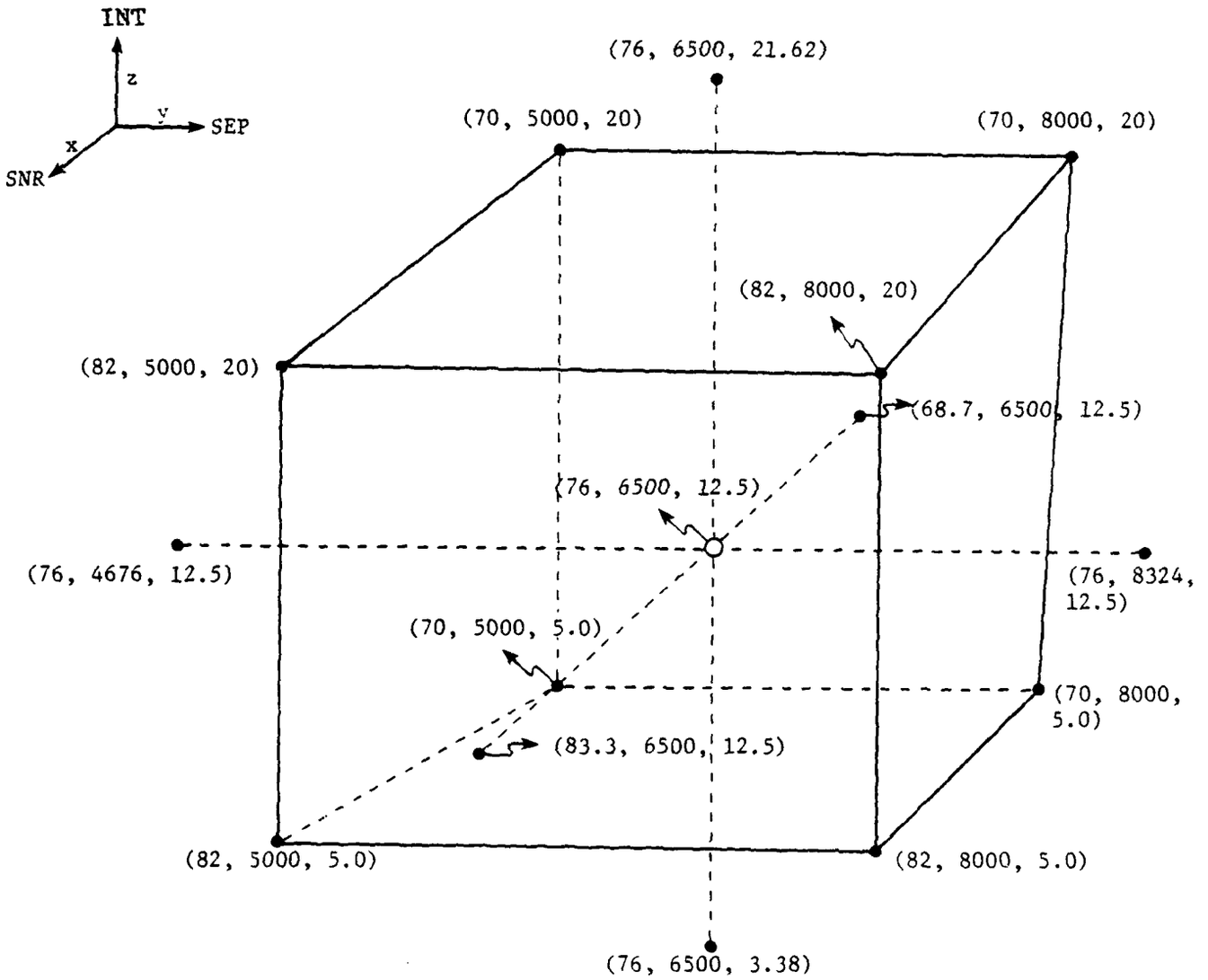


FIGURE A.2 - GEOMETRICAL REPRESENTATION OF THE REVISED OCCD

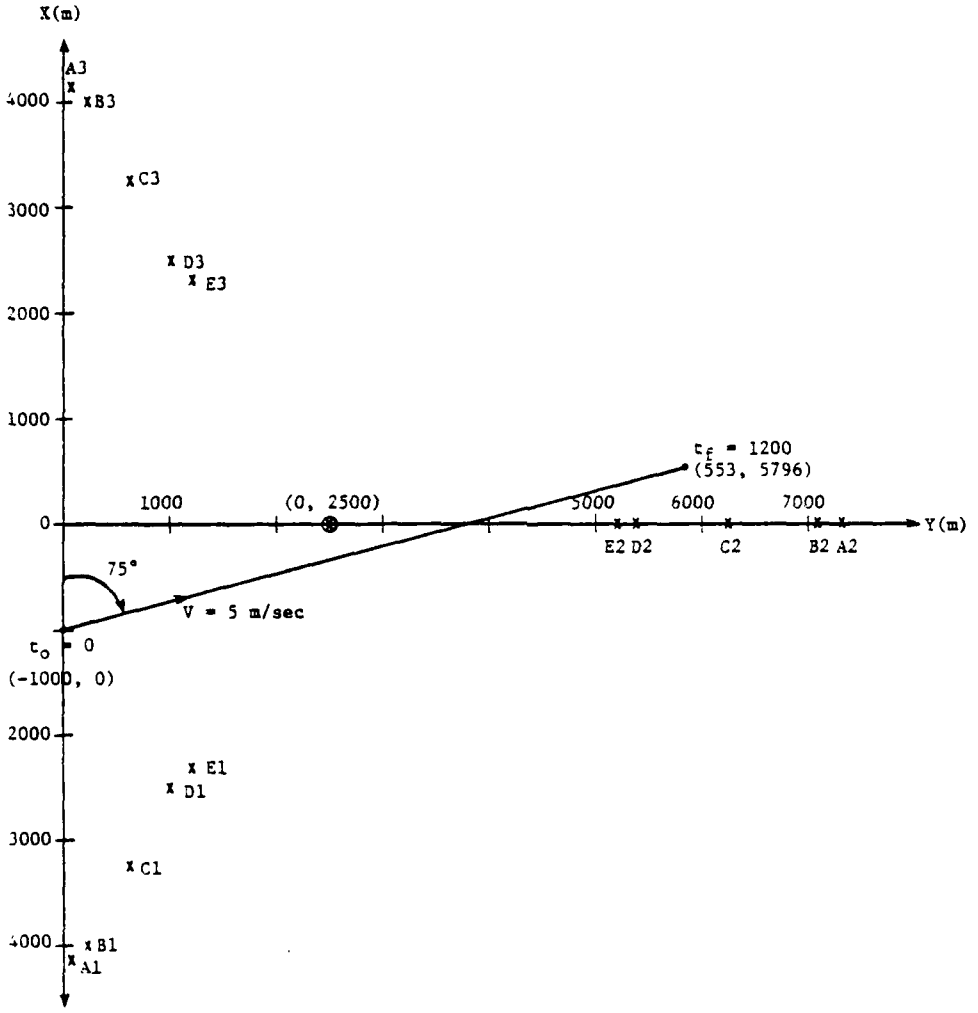


FIGURE A.3 - SCENARIOS FOR REVISED OCCD

TABLE A.III

DESCRIPTION OF REVISED OCCD'S SCENARIO

Initial Conditions for Target
t_o = 0 sec
x_o = -1000 m
y_o = 0 m
V_o = 5 m/sec
θ_o = 75°

Final Conditions for Target
t_f = 1200 sec
x_f = 553 m
y_f = 5796 m
V_f = 5 m/sec
θ_f = 75°

Centroid For All of the Tri-Tac Patterns
x = 0 m
y = 2500 m
V = 0 m/sec

TABLE A.III (Continued)

Buoy Positions for Revised OCCD					
Scenario	Design Level	SEP (m)	Buoy #	X (m)	Y (m)
A	+α	8,324	1	-4,162	97
			2	0	7,306
			3	4,162	97
B	+1	8,000	1	-4,000	191
			2	0	7,118
			3	4,000	191
C	0	6,500	1	-3,250	624
			2	0	6,252
			3	3,250	624
D	-1	5,000	1	-2,500	1,057
			2	0	5,386
			3	2,500	1,057
E	-α	4,676	1	-2,338	1,150
			2	0	5,200
			3	2,338	1,150

TABLE A.IV

RESPONSES FOR EACH TEST DESIGN LEVEL FOR THE REVISED OCCD

Case No.	Design Levels			Responses		
	SEP (m)	SNR (dB)	INT (sec)	CT (sec)	ADE (m)	PDE (m)
1	-1	-1	-1	112.50	273	677
2	-1	-1	+1	430.00	163	656
3	-1	+1	-1	32.50	61	383
4	-1	+1	+1	150.00	35	1015
5	+1	-1	-1	*1200.00	1053	2582
6	+1	-1	+1	370.00	451	589
7	+1	+1	-1	37.50	150	481
8	+1	+1	+1	130.00	65	104
9	0	0	0	218.75	297	404
10	$-\alpha$	0	0	81.25	78	94
11	$+\alpha$	0	0	*1200.00	2574	1602
12	0	$-\alpha$	0	581.25	299	878
13	0	$+\alpha$	0	168.75	87	234
14	0	0	$-\alpha$	45.63	218	869
15	0	0	$+\alpha$	140.53	104	206

*For these points, the Hybrid met its own convergence criteria, but the tracking errors were never reduced below 500 m. Rather than accept the output CT, the final time of 1200 sec. was assigned.

A.5 RSM Results and Analyses of These Results

This section and all further discussions describe the RSM results for the revised OCCD. Tables of the response surface fits, the optimization and eigenvalue results, as well as response surface contour plots are presented for each of the three responses.

A.5.1 Description of RSM Tables - Tables A.V, A.VII, and A.IX provide summaries of the analyses for the surfaces fit to the three response factors. Note that each of these three tables is divided into two sections. Descriptions of these sections follow:

(a) Response Surface Values and Statistics - This section contains information about the surface analyses of the statistical significance of the various estimated parameters which defined this surface.

(b) Response Surface Analysis - This section contains the information required to analyze the particular surface which has been fitted to the data.

Under section (a) the following pieces of information are given:

(1) Coded Betas - With RSM we were fitting a quadratic model of the form:

$$E(y) = B_o + \sum_{i=1}^{N} B_i X_i + \sum_{i=1}^{N} B_{ii} X_i^2 + \sum_{j=1}^{N-1} \sum_{i=j+1}^{N} B_{ji} X_j X_i$$

To prevent numerical problems encountered when inverting a matrix which contains values differing by several orders of magnitude and to eliminate correlation between the linear and quadratic terms in the model, the X_i's are coded variables of the form

$$X_i = \frac{Z_i - \mu_i}{\Delta_i} \quad \text{where}$$

Z_i = the raw data value

γ_i = the center point value

Δ_i = the distance from the center point to the +1 level of that variable in the factorial part of the OCCD

Thus, for SNR the coding is:

$$X_{SNR} = \frac{Z_{SNR} - 76}{6} ,$$

For SEP the coding is:

$$X_{SEP} = \frac{Z_{SEP} - 6500}{1500} , \text{ and}$$

For INT, it is:

$$X_{INT} = \frac{Z_{INT} - 12.5}{7.5} \quad .$$

Since there are zero correlations among all the coded variables, their squares, and their cross-products, the response of the dependent variable to a unit increase in one of the independent variables can be deduced directly from the model. Note, however, that a unit increase in the coded variables corresponds to an increase of Δ_i in the uncoded variables.

(2) Uncoded Betas - This column contains the coefficients of the quadratic surface expressed in uncoded form, that is, the model which uses these uncoded coefficients can use raw data to describe the response surface.

(3) F-Value - For each coefficient, an F-value is generated by computing the reduction in the total variance, caused by inclusion of this variable in the model, compared to the estimate of the variance of the process. This amounts to a test of hypothesis

$H_0 : B_k = 0$, versus

$H_1 : B_k \neq 0.$

If the F-value is greater than some critical value for a particular α-level, then we reject the hypothesis that the coefficient is zero and assume that, statistically, it is different from zero.

(4) α-Level - When testing any hypothesis, there are two kinds of errors which may be committed:

(a) Type 1 - To reject H_0 when it is actually true, and

(b) Type 2 - To accept H_0 when it is actually false.

The probability associated with Type 1 errors is called the size of the test and one minus the probability associated with the Type 2 error is called the power of the test. What would be most desirable is to both minimize the size and maximize the power of the test. Unfortunately, with a fixed sample size this cannot be done. Instead, the size of the test is fixed at some probability level, α, and the power is maximized. Thus when it is said that a hypothesis test is significant at an $\alpha = .1$ level, it is meant that the probability of Type 1 error has been fixed at 0.1 and the power of the test has been maximized (the probability of Type 2 error has been

minimized). An asterisk in the $\alpha = .1$ or $\alpha = .2$ columns means that the hypothesis test for this coefficient is significant at this α level.

Under section (b) the following information appears.

(1) One of the goals of RSM was to find an optimum or near optimum set of operating conditions for the response under consideration. Because the fitted surface was a quadratic, the usual techniques of multivariate calculus used to find stationary points was readily applied. This section gives the coordinates of the stationary point for this surface.

(2) Stationary Point Value - This gives the value of the fitted surface at the stationary point.

(3) Canonical Representation of Surface - In matrix representation, the estimated surface is given by

$$\hat{y}_0 = b_0 + \vec{b}^T\vec{x} + \vec{x}^T B\vec{x}$$

where

b_0 - is the intercept

$\vec{b}$ - is the vector $(\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_n)$ of estimates for the linear factors

$\vec{x}$ - is the vector $(x_1, \ldots, x_n)$

B - is the matrix

$$\begin{bmatrix} \hat{B}_{11} & \hat{B}_{12/2} & \cdots & \hat{B}_{1n/2} \\ \hat{B}_{12/2} & \hat{B}_{22} & \ddots & \hat{B}_{2n/2} \\ \vdots & & \ddots & \vdots \\ \hat{B}_{1n/2} & \cdots & & \hat{B}_{nn} \end{bmatrix}$$

Through a series of suitable translations and rotations, the equation above can be rewritten as:

$$\hat{y} = \hat{Y}_0 + \lambda_1 w_1^2 + \ldots + \lambda_n w_n^2$$

$$= \hat{Y}_0 + w^T \lambda w \text{ , where}$$

$\hat{Y}_0$ - is the value of the surface at the stationary point

$$\lambda = \begin{bmatrix} \lambda_1 & & o \\ & \ddots & \\ o & & \lambda_n \end{bmatrix}$$

where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of the B matrix

$w_1, \ldots, w_n$ are the coordinate axes in the eigenvalue system.

$$\vec{w} = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix} = M\vec{x}$$

where M is a matrix consisting of the normalized eigenvectors of B.

There are several things which can be determined from the canonical representation of the response surface:

(a) If all λ_i's are negative, we have a maximum point, if they are positive we have a minimum point, if they are both positive and negative we have a saddle point;

(b) The λ_i's indicate the directions of greatest increase or decrease of the response in terms of the w_i coordinate axes;

(c) This representation helps to determine the shape and characteristics of the response surface so that nearly optimum operating conditions can be determined.

(4) x to w Transformations - This is the set of linear equations that relates the x-coordinates to the w-coordinates.

Thus, when a particular operating point or set of operating conditions is determined by using the canonical form of the equations, this transformation can be used to find the appropriate set of x-values.

A.5.2 RSM Analysis of the Hybrid's ADE Response

Table A.V describes the response surface information for the average distance error (ADE) data. Residual versus fitted ADE plots, generated by the raw ADE values, indicated that as ADE increased, the variance of the residuals increased. This is the usual indication that a log transformation should be applied to the dependent variable. The log transformation succeeded in reducing the variance of the residual sum of squares after the fit, so response surfaces of the following form were generated:

$$\log_{10}(\text{ADE}) = b_0 + \vec{b}^T\vec{x} + \vec{x}^T B\vec{x} \ .$$

The regression F-value indicates a regression which is significant at the $\alpha = .1$ level and the R^2 value shows that, after taking the mean into account, the surface accounts for about 88% of the remaining variance in the data. The F-values for the coefficients indicate that, by far, the most significant factors are the linear SEP and SNR terms. Of somewhat less importance are the linear and quadratic INT terms.

The analysis of this surface indicates that there is a stationary point just outside the experimental region with an ADE value of 146.0. The eigenvalues show that the stationary point is a saddle point with the directions of maximum decrease along the w_1 and w_2 axes. From the x to w

TABLE A.V

RSM RESULTS FOR LOG_{10} (ADE)

RESPONSE SURFACE VALUES & STATISTICS

VARIABLE	CODED BETA	UNCODED BETA	F-VALUE	$\alpha = .1$	$\alpha = .2$
$\hat{\beta}_0$	2.28	-25.166			
$\hat{\beta}_1$ (SEP)	.322	- .000029	13.4	*	*
$\hat{\beta}_2$ (SNR)	- .334	.755	14.4	*	*
$\hat{\beta}_3$ (INT)	- .145	.0939	2.7		*
$\hat{\beta}_{11}$	.117	$.519 \times 10^{-7}$	.7		
$\hat{\beta}_{22}$	- .184	- .00511	1.7		
$\hat{\beta}_{33}$	- .205	- .00364	2.2		*
$\hat{\beta}_{12}$	- .047	$-.518 \times 10^{-5}$	.2		
$\hat{\beta}_{13}$	- .034	$-.298 \times 10^{-5}$	.1		
$\hat{\beta}_{23}$	- .0017	$-.383 \times 10^{-4}$	0.0		

Regression F - 3.95 - Significant at $\alpha = .1$ $R^2 = 87.7$

RESPONSE SURFACE ANALYSIS

Stationary Point Coordinates:

SEP = 4171.7 SNR = 71.7 INT = 10.8

Stationary Point Value:

146.0

Eigenvalues:

$\lambda_1 = - .00511$ $\lambda_2 = - .00364$ $\lambda_3 = 5.38 \times 10^{-8}$

TABLE A.V (Cont.)

X TO W TRANSFORMATION

$$\begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} = \begin{bmatrix} -.000511 & -1.0 & .013 \\ .0004 & -.013 & 1.0 \\ -1.0 & .00051 & .00041 \end{bmatrix} \begin{bmatrix} SEP \\ SNR \\ INT \end{bmatrix} + \begin{bmatrix} 74.0 \\ -11.55 \\ 4171.7 \end{bmatrix}$$

transformation, it can be seen that w_1 direction corresponds basically to the SNR and the w_2 direction corresponds basically to INT. Thus ADE can be reduced by increasing SNR and INT. Also, from the x to w transformation it can be seen that w_3 corresponds to SEP and the associated eigenvalue indicates that decreasing sensor separation also decreases ADE. Note, however, that all of the λ_i's are quite small, which indicates a fairly flat surface for the quadratic response.

Figure A.4 and A.5 show the three-dimensional plot and contour plot, respectively, for the log (ADE) response for SNR = 68.704 dB. Table A.VI defines the values for the contour symbols used in Figure A.5. The contour plot indicates that the lowest ADE values occur for small separation distances and long update intervals. There are two primary reasons for this:

(1) Due to the previously described inverse relationship, long integration times correspond to very accurate data estimates.

(2) Once initialization has occurred, highly accurate data measurements lead to more accurate state vector estimates and thus lower the distance errors.

However, as separation distances increase, the algorithm becomes less and less sensitive to integration time. For large separation distances, the same ADE occurs for the entire range of INT's. This is especially true at low SNR values.

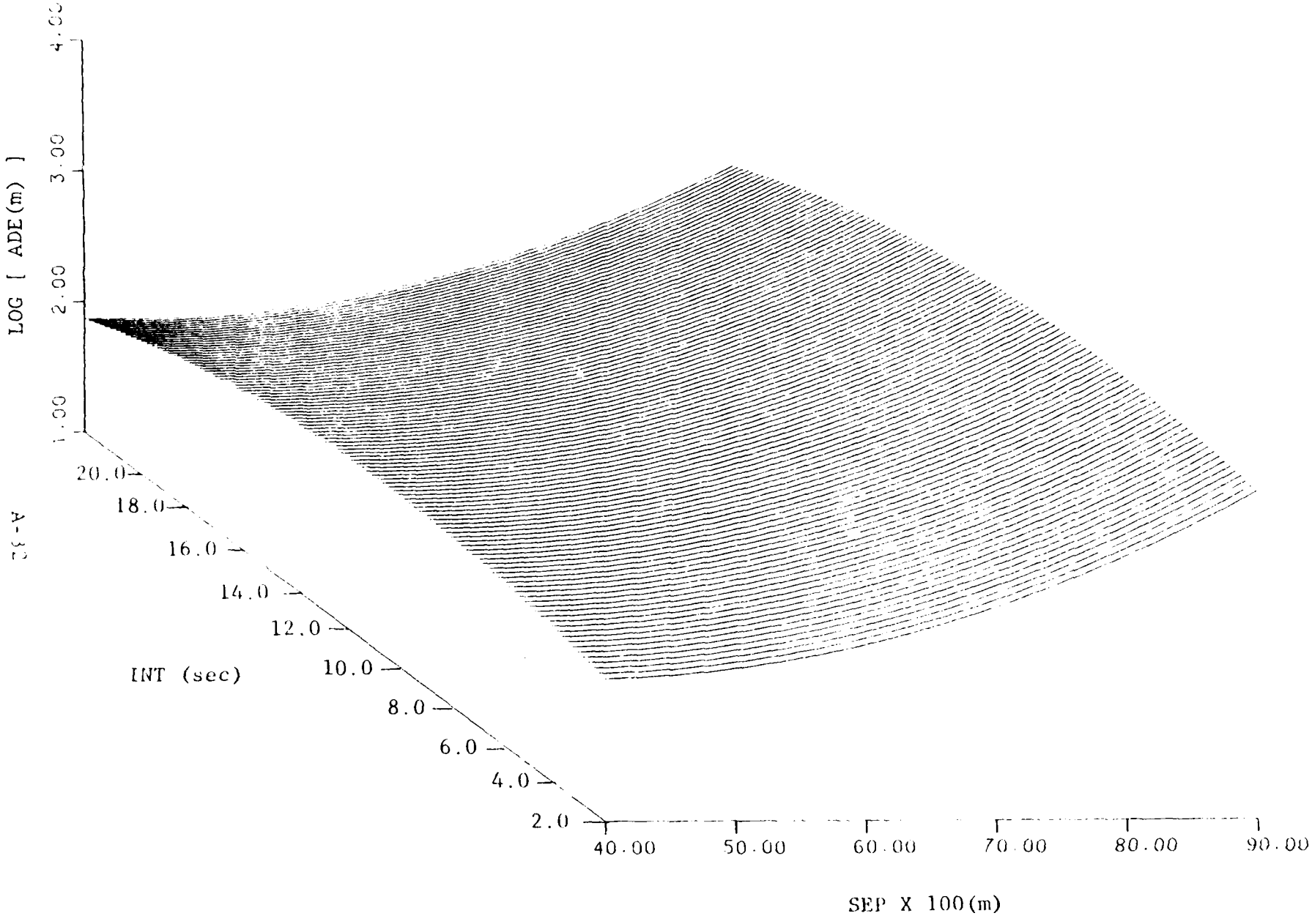


Figure A.4 RESPONSE SURFACE PLOT OF LOG_{10} (ADE) FOR SNR = 68.704

TABLE A.VI

DEFINITION OF CONTOUR

SYMBOLS FOR LOG_{10} (ADE)

Symbol	Contour Value
1	1.25
2	1.50
3	1.75
4	2.00
5	2.25
6	2.50
7	2.75
8	3.00
9	3.25
A	3.50

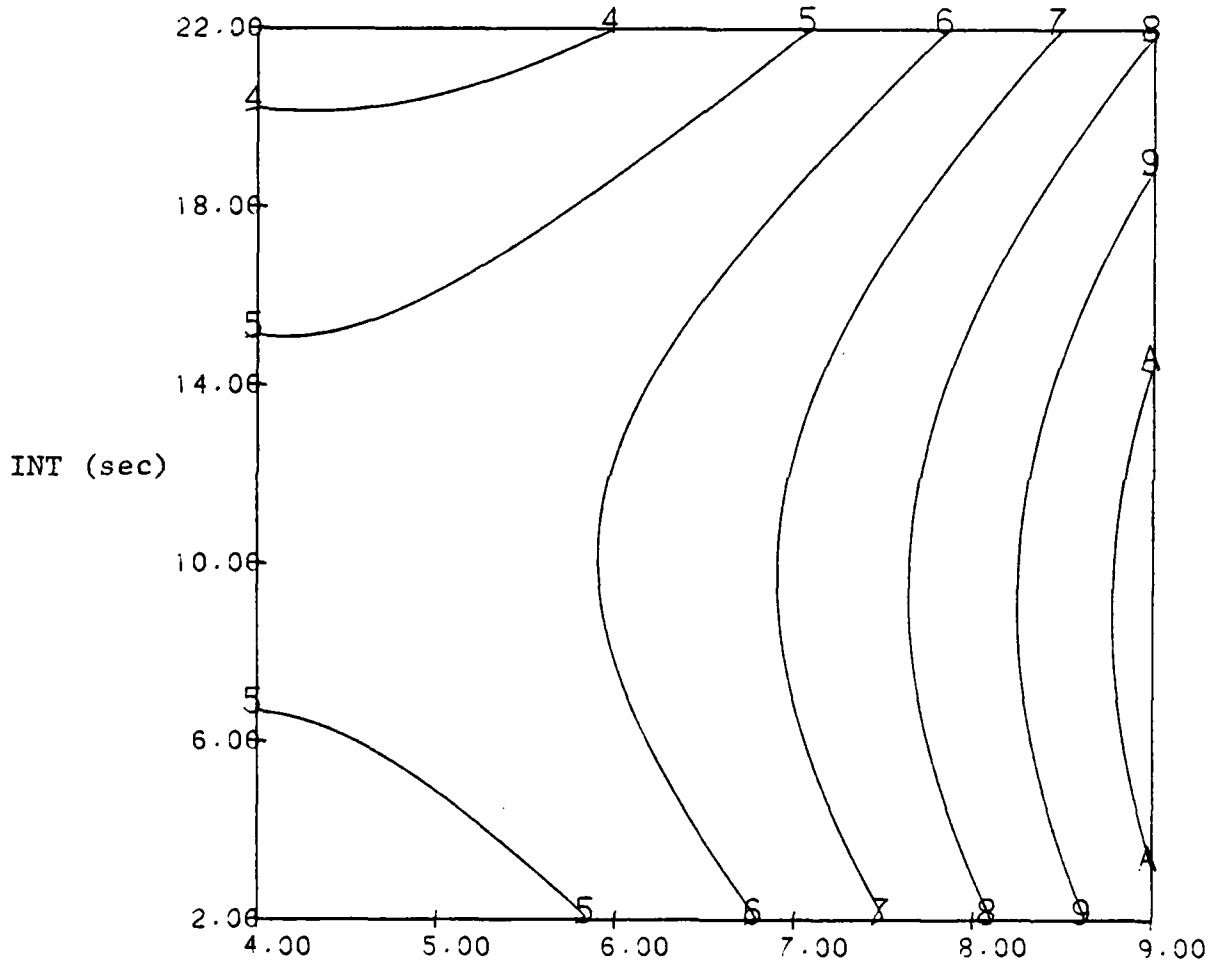


Figure A.5 CONTOUR PLOT OF LOG_{10} (ADE) FOR SNR = 68.704

A.5.3 RSM Analysis of the Hybrid's CT Response

Table A.VII contains the response surface information for the convergence time (CT) data. As with the ADE data, residual versus predicted CT plots generated by the raw CT values indicated that as CT increased, the variance of the residuals increased. A log transformation was applied to CT and a model of the form

$$\log_{10}(CT) = b_0 + \vec{b}^T\vec{x} + \vec{x}^T B\,\vec{x}$$

was fitted to the data. The regression F-value was significant at the $\alpha = .1$ level and the R^2 value indicates that, after adjusting for the mean, the surface accounts for about 87% of the remaining variance in the data. From the coefficient F-values, it can be seen that the important terms in the model are linear SEP and SNR terms and, to a somewhat lesser degree, the linear and quadratic INT terms.

The analysis of this surface indicates that there is a saddle point which lies just outside of the experimental region and the value of the function at this point is 113.9. From the canonical representation, it can be seen that the surface decreases along the w_1 axis and increases along the w_2 and w_3 axes. The w to x transformation shows that the w_2 axis corresponds to SEP, while the w_1 and w_3 axes are made up of both SNR and INT contributions. Figure A.6 illustrates the 3-dimensional plot generated by fixing SNR at its lowest level(68.704 dB), and then graphing the resulting equations as functions of SEP and INT. Figure A.7 is the contour plot associated with this graph. Table A.VIII contains a table of the values for the various contour lines. From the contour plots, it is clear that minimum CT's occur for small values of SEP and INT. Furthermore, as SNR increases, SEP

TABLE A. VII

RSM RESULTS FOR LOG_{10} (CT)

RESPONSE SURFACE VALUES & STATISTICS

VARIABLE	CODED BETA	UNCODED BETA	F-VALUE	$\alpha = .1$	$\alpha = .2$
$\hat{\beta}_0$	2.26	14.94			
$\hat{\beta}_1$ (SEP)	.218	.000844	5.8	*	*
$\hat{\beta}_2$ (SNR)	- .329	- .366	13.2	*	*
$\hat{\beta}_3$ (INT)	.171	.0118	3.5		*
$\hat{\beta}_{11}$	.084	$.374 \times 10^{-7}$	.3		
$\hat{\beta}_{22}$	.085	.00236	.4		
$\hat{\beta}_{33}$	- .316	- .00561	4.8	*	*
$\hat{\beta}_{12}$	- .120	$- .134 \times 10^{-4}$	1.3		
$\hat{\beta}_{13}$	- .152	$- .135 \times 10^{-4}$	2.1		
$\hat{\beta}_{23}$	.142	.00315	1.8		

Regression F - 3.68 - Significant at $\alpha = .1$ $R^2 = 86.9$

RESPONSE SURFACE ANALYSIS

Stationary Point Coordinates:

SEP = 7302.1 SNR = 87.1 INT = 16.7

Stationary Point Value:

113.9

Eigenvalues:

$\lambda_1 = -.00591$ $\lambda_2 = 1.93 \times 10^{-8}$ $\lambda_3 = .00266$

TABLE A. VII (Cont.)

X TO W TRANSFORMATION

$$\begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} = \begin{bmatrix} .000913 & -.187 & .982 \\ -1.0 & -.0031 & .00035 \\ .00294 & -.982 & -.187 \end{bmatrix} \begin{bmatrix} SEP \\ SNR \\ INT \end{bmatrix} + \begin{bmatrix} - 6.77 \\ 7302.8 \\ 67.2 \end{bmatrix}$$

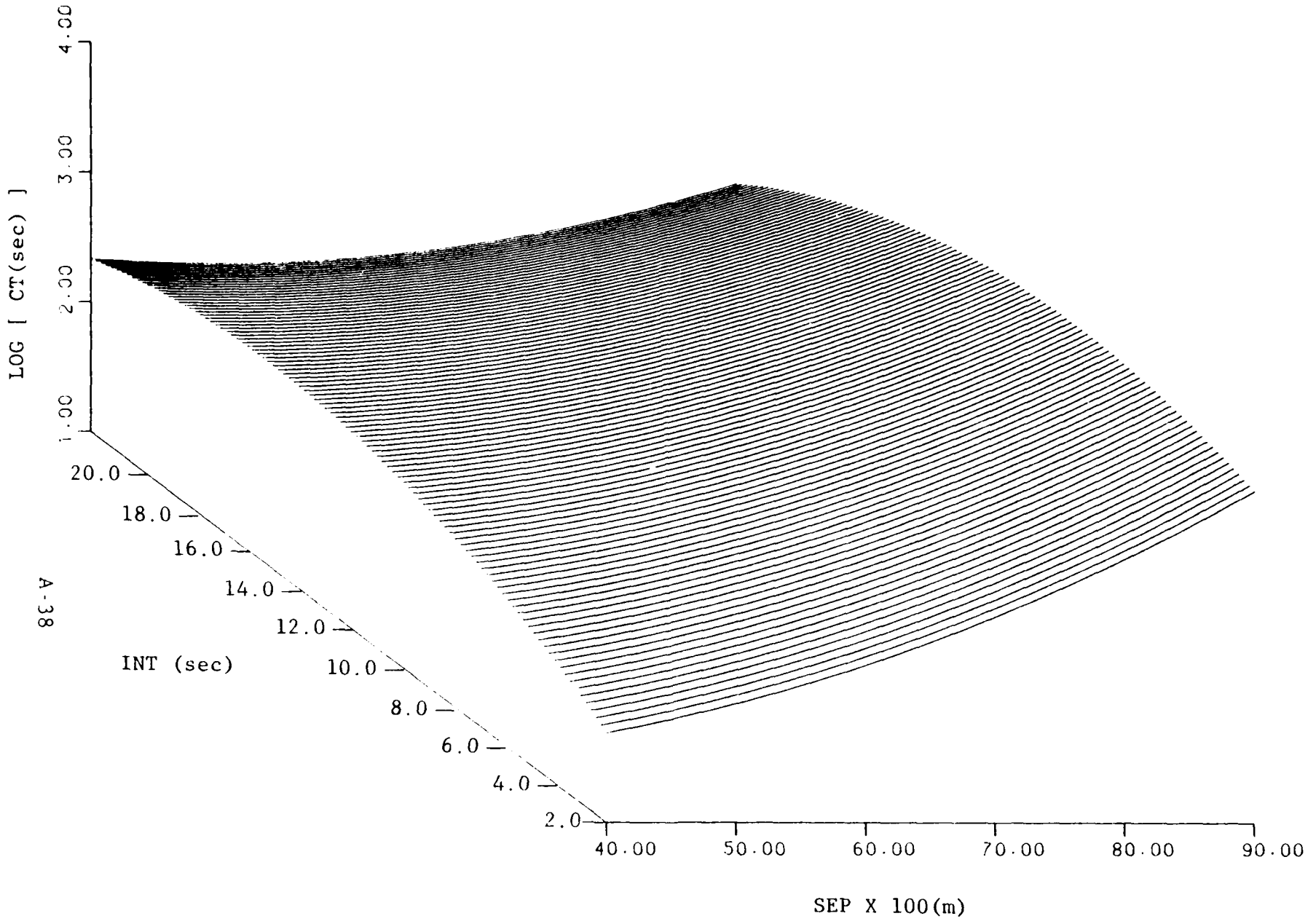


Figure A.6 RESPONSE SURFACE PLOT OF LOG_{10} (CT) for SNR = 68.704

TABLE A. VIII

DEFINITION OF CONTOUR

SYMBOLS FOR LOG_{10} (CT)

Symbol	Contour Value
1	1.00
2	1.25
3	1.50
4	1.75
5	2.00
6	2.25
7	2.50
8	2.75
9	3.00
A	3.25
B	3.50
C	3.75
D	4.00

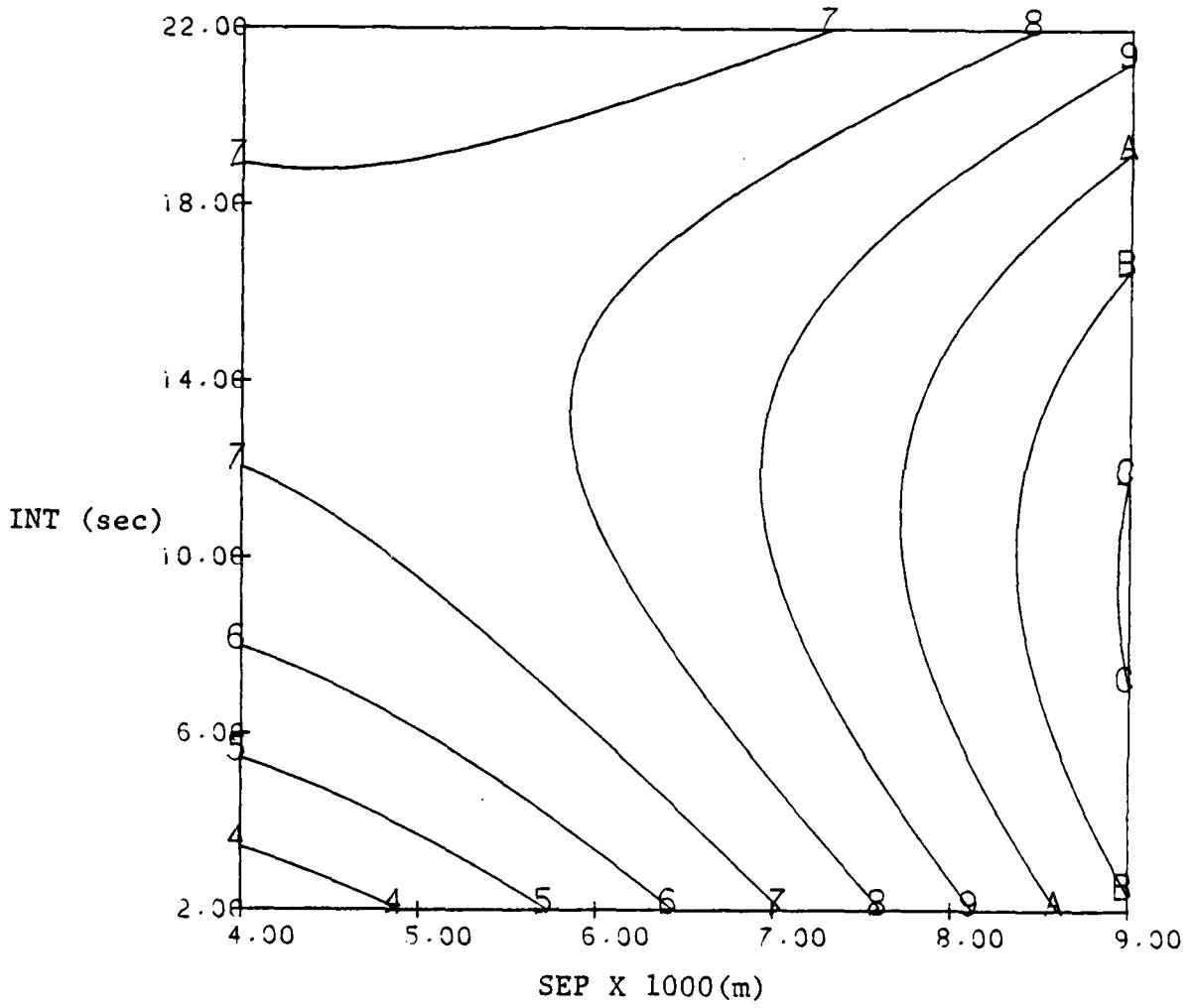


Figure A.7 CONTOUR PLOT OF LOG_{10} (CT) FOR SNR = 68.704

becomes less and less important while INT retains its importance.

A.5.4 RSM Analysis of the Hybrid's PDE Response - Table A.IX contains the response surface information for the predicted distance error (PDE) data. Residual plots indicated that no transformation of the independent variable was necessary, so the fitted model is of the form

$$PDE = b_0 + \vec{b}^T\vec{x} + \vec{x}^T B\vec{x}.$$

The regression F-value indicates significance at the $\alpha = .1$ level and the R^2 value is 89.0. The coefficient F-values indicate that all of the linear terms are significant, the quadratic SEP is significant, and all the two-factor interaction terms are significant. The eigenvalues for this system indicate that the extremum for this response is a saddlepoint that lies outside the test region.

The three-dimensional and contour plots for PDE indicate several interesting things. (The contour symbol values are presented in Table A.X) First, for low SNR values as are shown in Figures A.8 and A.9, there is a very rapid degradation in PDE as SEP increases. The optimal situation occurs for highly accurate data measurement (large INT values) with small to moderate SEP values. Thus, large areas of sensor overlap in conjunction with high quality data can assure low PDE values for low SNR values. Secondly, for a SNR value of 76.0 dB, Figures A.10 and A.11 show that there is a wider range of optimal values. Roughly, as SEP increases, INT must also increase to keep PDE at a minimum value. Thirdly, for the high

TABLE A.IX

RSM RESULTS FOR PDE

RESPONSE SURFACE VALUES & STATISTICS

VARIABLE	CODED BETA	UNCODED BETA	F-VALUE	$\alpha = .1$	$\alpha = .2$
$\hat{\beta}_0$	718.36	3282.57			
$\hat{\beta}_1$ (SEP)	260.91	1.9	5.71	*	*
$\hat{\beta}_2$ (SNR)	-301.65	- 144.27	7.6	*	*
$\hat{\beta}_3$ (INT)	-234.12	- 316.33	4.6	*	*
$\hat{\beta}_{11}$	257.43	.00011	2.2		*
$\hat{\beta}_{22}$	60.26	1.67	.12		
$\hat{\beta}_{33}$	47.71	.848	.08		
$\hat{\beta}_{12}$	-331.26	- .036	6.7	*	*
$\hat{\beta}_{13}$	-372.69	- .033	8.5	*	*
$\hat{\beta}_{23}$	283.76	6.31	5.0	*	*

Regression F - 4.5 - Significant at $\alpha = .1$ $R^2 = 89.0$

RESPONSE SURFACE ANALYSIS

Stationary Point Coordinates:

SEP = 3251.61 SNR = 65.31 INT = 7.18

Stationary Point Value

787.41

Eigenvalues:

$\lambda_1 = -1.919$ $\lambda_2 = -2.36 \times 10^{-5}$ $\lambda_3 = 4.441$

TABLE A. IX (Cont.)

W TO X TRANSFORMATION

$$\begin{bmatrix} W_1 \\ W_2 \\ W_3 \end{bmatrix} = \begin{bmatrix} -.0002 & .66 & -.752 \\ -1.0 & -.0043 & -.0036 \\ .0056 & -.752 & -.66 \end{bmatrix} \begin{bmatrix} SEP \\ SNR \\ INT \end{bmatrix} + \begin{bmatrix} -37.16 \\ 3251.9 \\ 35.7 \end{bmatrix}$$

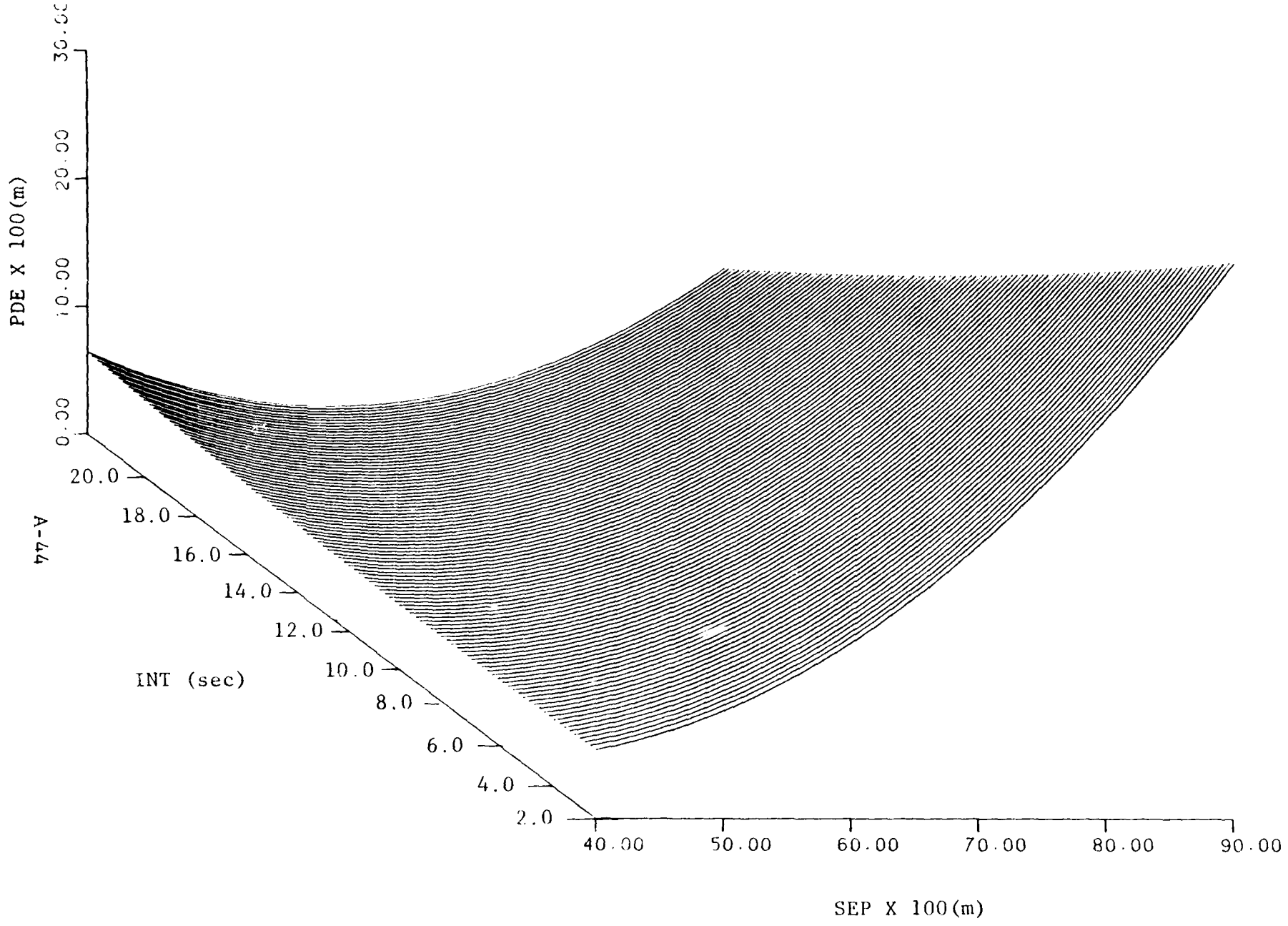


Figure A.8 - RESPONSE SURFACE PLOT OF PDE FOR SNR = 68.704

TABLE A. X

DEFINITION OF CONTOUR

SYMBOLS FOR PDE

Symbol	Contour Value
1	200
2	400
3	600
4	800
5	1000
6	1200
7	1400
8	1600
9	1800
A	2000
B	2500
C	3000
D	3500
E	4000

AD-A118 470 TRACOR APPLIED SCIENCES AUSTIN TX F/G 15/4
HYBRID TRACKING ALGORITHM IMPROVEMENTS AND CLUSTER ANALYSIS MET--ETC(U)
FEB 82 D COOPER, G CORSER, T WILSON N00014-78-C-0670
UNCLASSIFIED TRACOR-T82-AU-9054-U NL

3 OF 3
AD A
118470

END
DATE FILMED
10-82
DTIC

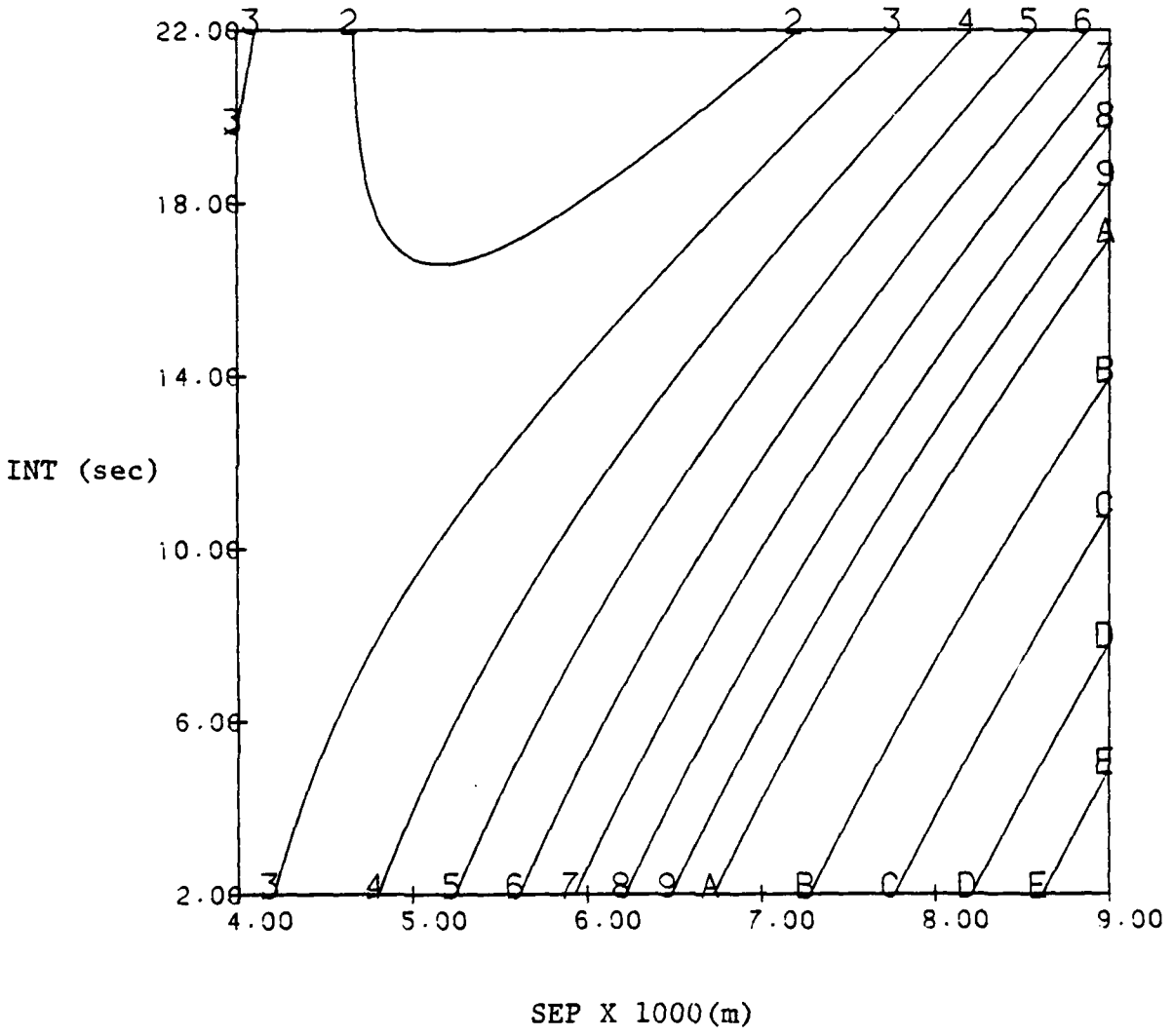


Figure A.9 - CONTOUR PLOT OF PDE FOR SNR = 68.704

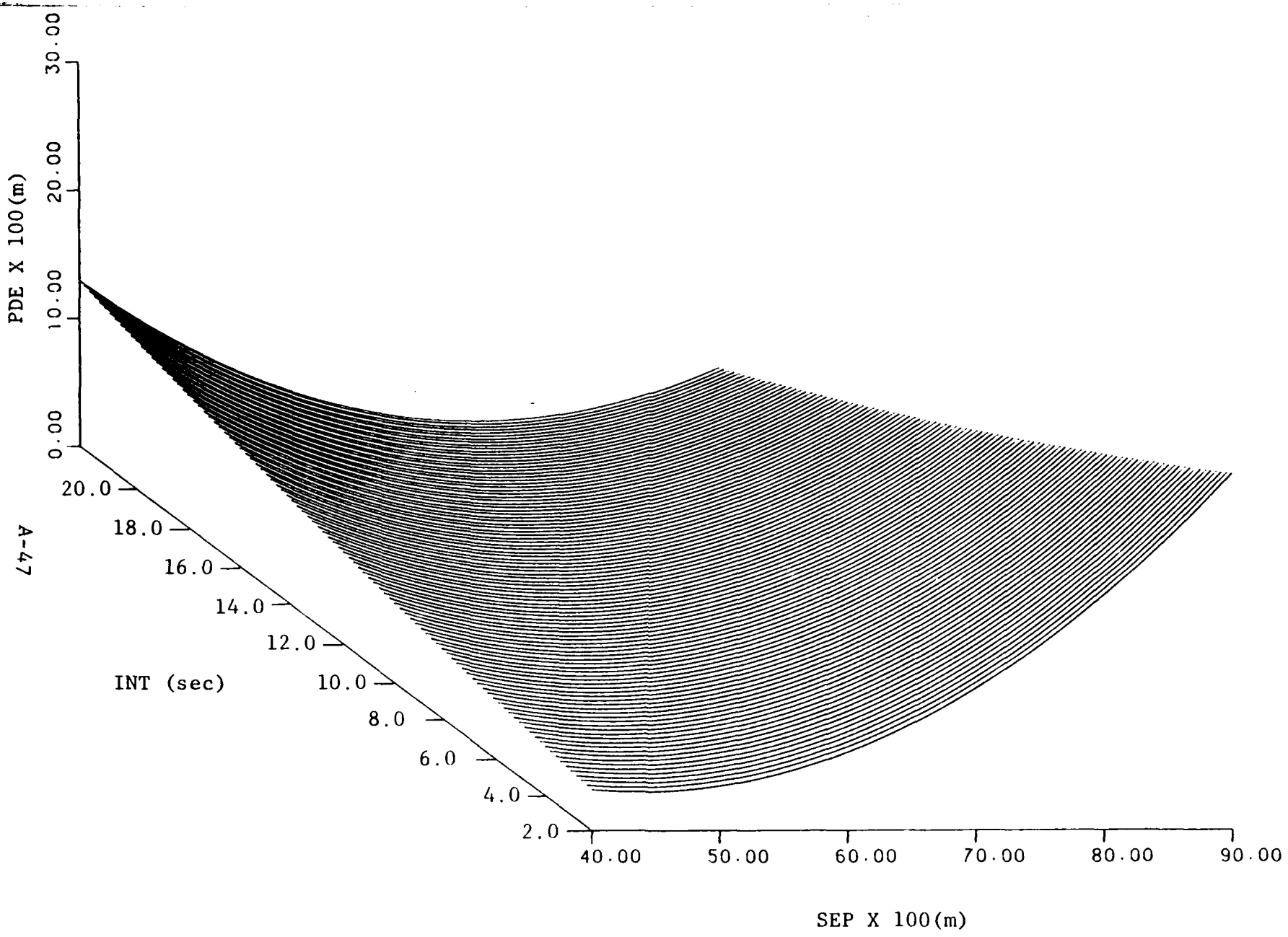


Figure A.10 - RESPONSE SURFACE PLOT OF PDE FOR SNR = 76.0

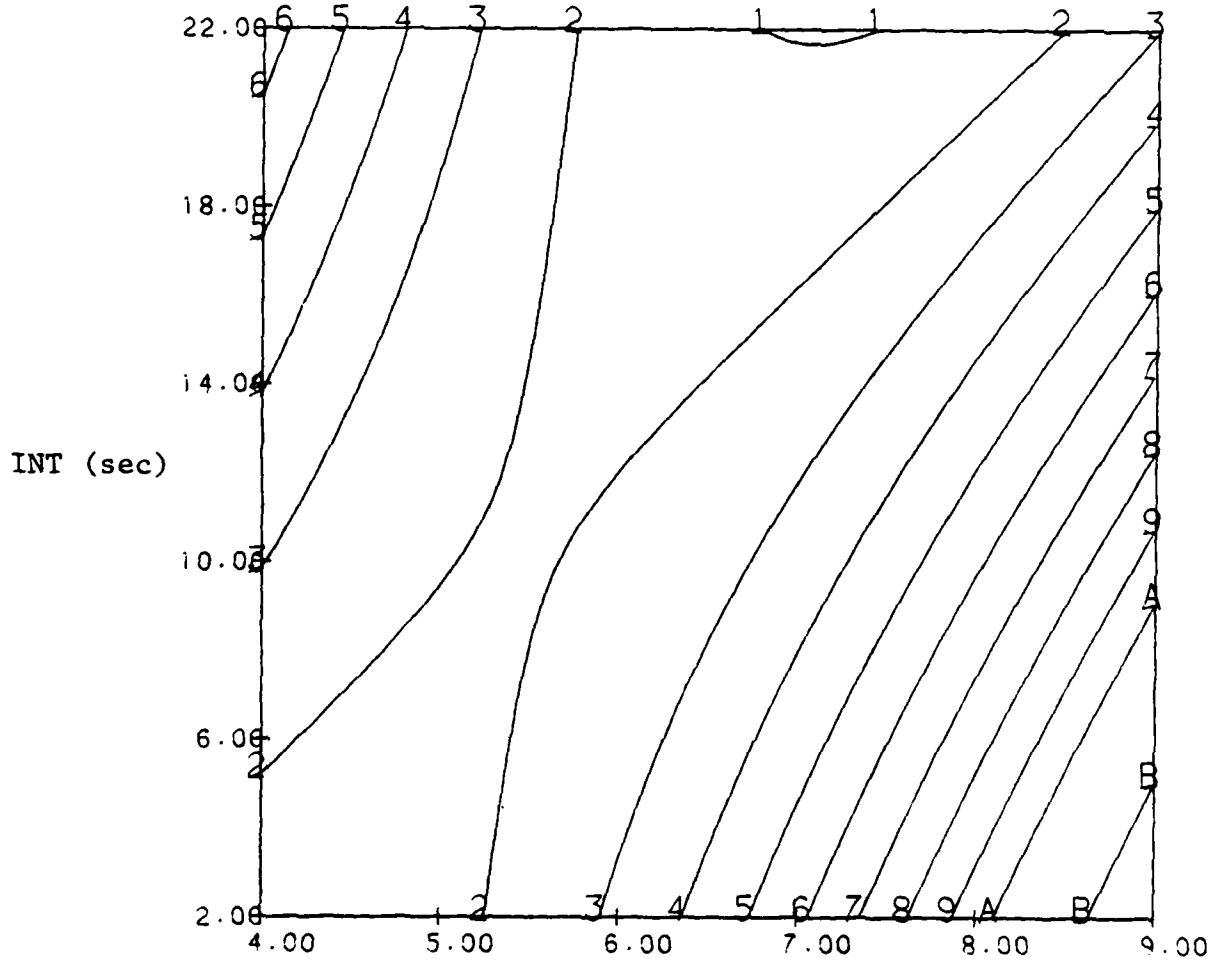


Figure A.11 - CONTOUR PLOT OF PDE FOR SNR = 76.0

SNR value of 83.296 dB, Figures A.12 and A.13 illustrate that there is a wide range of conditions for minimizing PDE. Essentially, as SEP increases, INT must increase also.

The interpretation of significant two-factor interactions can be seen from Figure A.11 which shows the contour plot of the surface that results from fixing SNR at its average (= 76.0) value and allowing SEP and INT to wander over their respective test ranges. Note that at low values for INT, PDE goes high to low as SEP increases, while at high INT values, PDE goes from low to high values as SEP increases. If the two curves were plotted on the same axes, a pair of intersecting parabolas would result. Thus, the meaning of a significant SEP-INT interaction is that for the average SNR value, PDE behaves quite differently for changes in SEP at low INT values than it does for changes in SEP at high INT values. The same reasoning applies to significant SEP-SNR and SNR-INT interactions.

Thus, to minimize PDE there must be high quality data for large buoy separation distances. For small sensor separation distances, there must be a great deal of data with quality being of less importance. The ranges over which these statements apply vary, of course, with SNR.

A.6 Conclusions From This RSM Study

This RSM study has quantified Hybrid's performance measures ADE, CT, and PDE as functions of three independent factors -- SEP, SNR and INT. It must be emphasized that these results were valid only for this particular scenario and for those values of the independent factors that fell within our test region. This particular scenario was for a

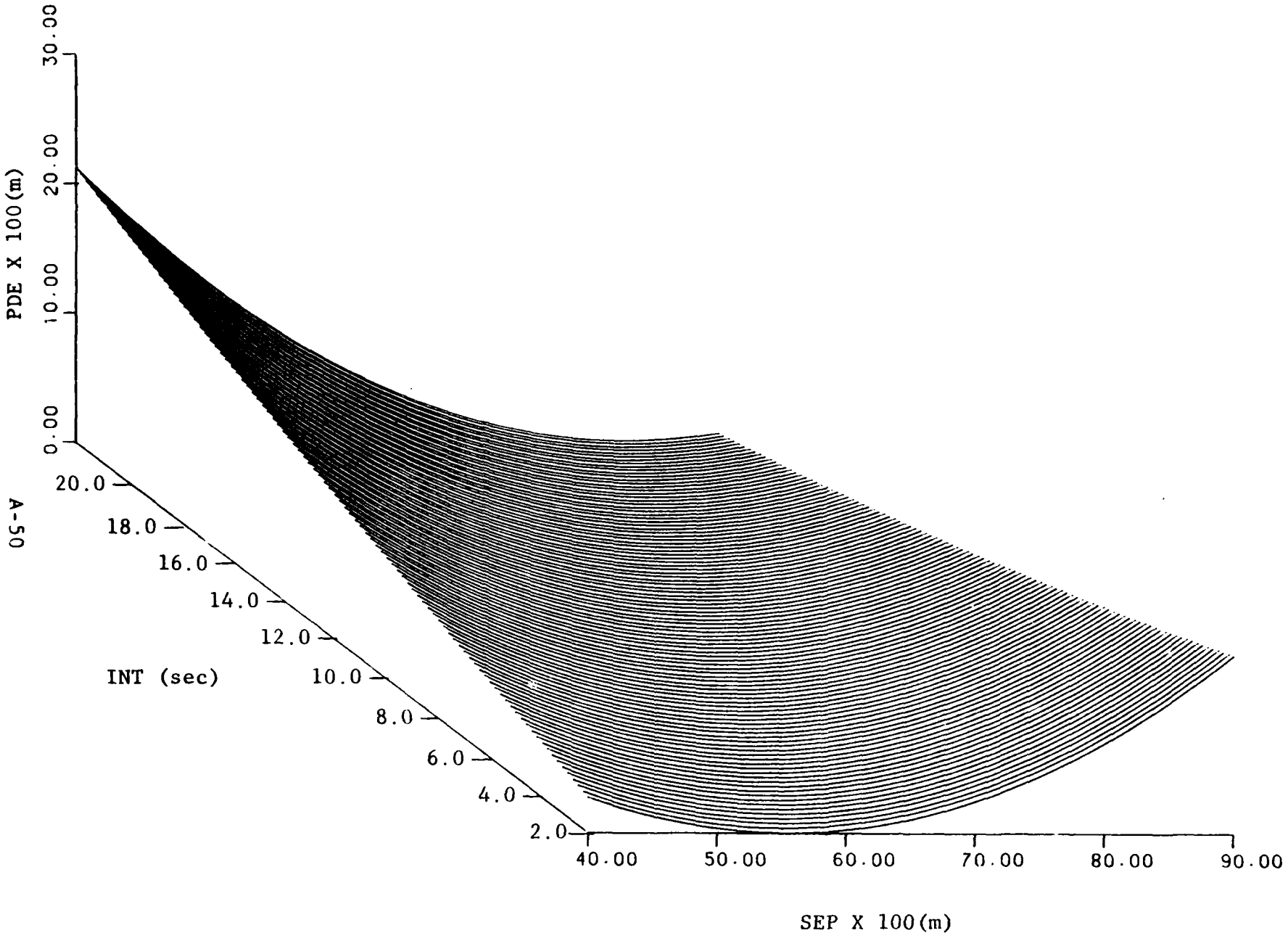


Figure A.12 RESPONSE SURFACE PLOT OF PDE FOR SNR = 83.296

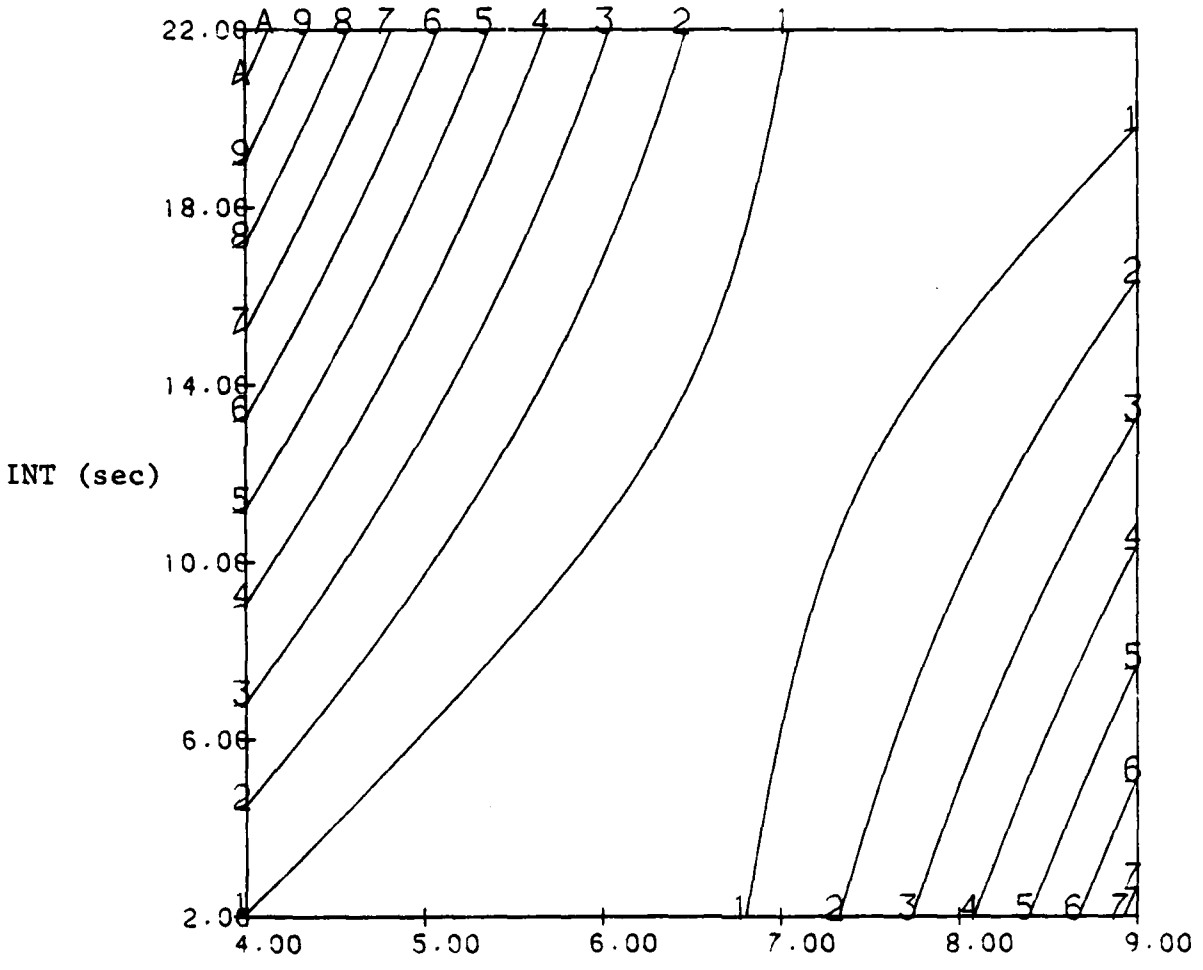


Figure A.13 CONTOUR PLOT OF PDE FOR SNR = 83.296

nonmaneuvering target whose trajectory ran through the tri-tac, sonobuoy pattern. The response surfaces could very well be different for targets that used a maneuvering trajectory or that used a trajectory that ran outside the sonobuoy field. For each of the three Hybrid responses, their respective extremum points fell outside the test region. These extrema responses may not be accurate because the error of values extrapolated outside the test region may increase rapidly. Nevertheless, when one understands the limitations of this approach, RSM techniques prove to be very useful for quantifying Hybrid's response to the data gathering factors SEP, SNR and INT. The results from this study appear to be credible because they can be explained intuitively as they were in previous subsections. The RSM approach has been a very useful tool for examining Hybrid's tracking performance and can be useful for examining the response of it and other trackers in the future.

DISTRIBUTION LIST

FOR

Hybrid Passive Tracking Algorithm Improvements and Cluster Analysis Methods

Statistics and Probability
Program (Code 411(SP))
Office of Naval Research
Arlington, VA 22217 (3 copies)

Naval Ocean Systems Center
Code 6212
San Diego, CA 92152

Naval Surface Weapons Center
White Oak Laboratory
Code U-20
Silver Spring, MD 20910 (2 copies)

Mr. J. Conrad
Naval Intelligence Support Center
Code 20C
Suitland, MD 20390

Naval Air Development Center
Warminster, PA 18974

Naval Electronic Systems Command
Code 612, Code 615
Washington, DC 20360

Naval Research Laboratory
Code 2627, Code 5308, Code 7932
Washington, DC 20375

Naval Sea Systems Command
Code 63R-1, Code 63R-16
Washington, DC 20360

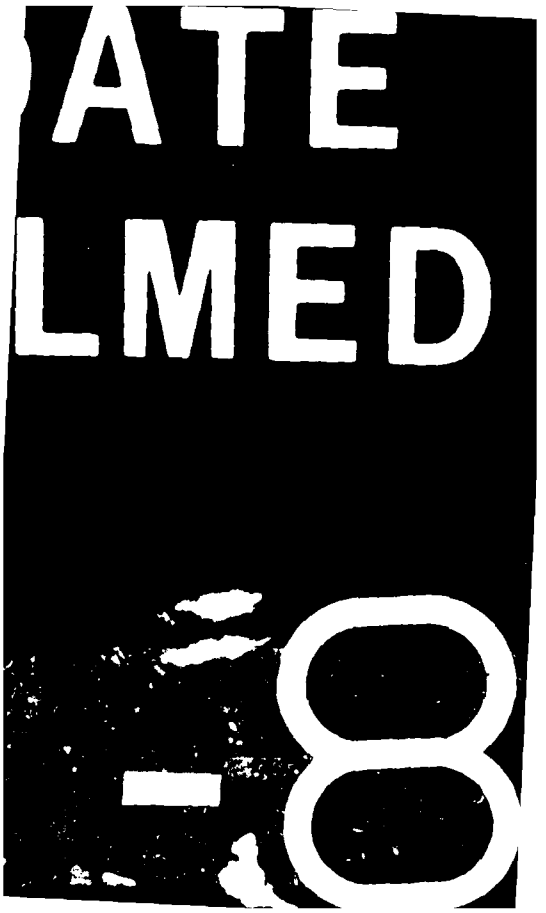
Naval Air Systems Command
Code 370, Code 310
Jefferson Plaza No. 1
Arlington, VA 20360

Naval Underwater Systems Center
Code 352, Code 351
Newport, RI 02840

Defense Technical Information
Center
Cameron Station
Alexandria, VA 22314 (12 copies)

Dr. Byron D. Tapley
Department of Aerospace
Engineering and Engineering
Mechanics
The University of Texas at Austin
Austin, Texas 78712

Dr. C. Carter
Code 313
Naval Underwater Systems Center
New London Laboratory
New London, CT 06320


ATE
LMED
8